Formulación y Resolución de Modelos de Programación Matemática en Ingeniería y Ciencia.

Enrique Castillo, Antonio J. Conejo, Pablo Pedregal,
Ricardo García y Natalia Alguacil

20 de febrero de 2002

DEDICATORIA

A Alfonso Fernández Canteli
por su constante ánimo y apoyo

Índice General

Prefacio **xi**

I Modelos 1

1 Programación lineal **3**
1.1 Introducción . . . 3
1.2 El problema del transporte . . . 4
1.3 El problema de la planificación de la producción . . . 6
1.4 El problema de la dieta . . . 9
1.5 El problema del flujo en una red . . . 11
1.6 El problema de la cartera de valores . . . 13
1.7 El sistema de vigas y cuerdas . . . 15
1.8 El problema del despacho económico . . . 18
Ejercicios . . . 22

2 Programación lineal entera-mixta **25**
2.1 Introducción . . . 25
2.2 El problema de la mochila . . . 25
2.3 Identificación de síntomas relevantes . . . 27
2.4 El problema de la Academia de Ingeniería . . . 29
2.5 El problema del horario . . . 32
2.6 Modelos de localización de plantas productivas . . . 35
2.7 Programación de centrales térmicas de producción de electricidad 39
Ejercicios . . . 44

3 Programación no lineal **47**
3.1 Introducción . . . 47
3.2 Algunos ejemplos geométricos . . . 47
3.2.1 El paquete postal . . . 47
3.2.2 La tienda de campaña . . . 48
3.2.3 La bombilla . . . 48
3.2.4 La superficie . . . 50
3.2.5 El transporte de arena . . . 50

3.3 Algunos ejemplos mecánicos 51
3.3.1 El voladizo 51
3.3.2 La estructura de dos barras 52
3.3.3 La columna sometida a pandeo 53
3.3.4 El sistema de vigas y cuerdas 54
3.4 Algunos ejemplos de ingeniería eléctrica 56
3.4.1 Estimación de estado en sistemas eléctricos 56
3.4.2 Reparto óptimo de carga 58
3.5 El problema de la matriz equilibrada 62
3.6 El problema de la asignación de tráfico 66
Ejercicios 69

II Métodos 73

4 Introducción a la programación lineal 75
4.1 Introducción 75
4.2 Formulación del problema 75
4.3 Problema de programación lineal en forma estándar 80
4.3.1 Transformación a la forma estándar 81
4.4 Soluciones básicas 83
4.5 Sensibilidades 85
4.6 Dualidad 86
4.6.1 Obtención del dual a partir del primal en forma estándar 87
4.6.2 Obtención del problema dual 88
4.6.3 Teoremas de dualidad 89
Ejercicios 95

5 El conjunto de soluciones factibles 99
5.1 Introducción y motivación 99
5.2 Conjuntos convexos 101
5.3 Espacios vectoriales 108
5.4 Conos poliédricos convexos 109
5.5 Politopos 112
5.6 Poliedro 113
5.7 PPL acotado y no acotado 115
Ejercicios 116

6 Resolución de problemas de programación lineal 119
6.1 Introducción 119
6.2 El método simplex 120
6.2.1 Ejemplo ilustrativo 120
6.2.2 Descripción general 122
6.2.3 Etapa de iniciación 123
6.2.4 Operación elemental de pivotación 125
6.2.5 Identificación de una solución óptima 126

6.2.6 Iteración reguladora 128
6.2.7 Detección de no acotación 128
6.2.8 Detección de no factibilidad 129
6.2.9 Etapa de iteraciones estándar 129
6.2.10 El algoritmo simplex revisado 131
6.2.11 Algunos ejemplos ilustrativos 133
6.3 El método del punto exterior 143
6.3.1 Fase inicial 145
6.3.2 Fase reguladora 146
6.3.3 Detección de infactibilidad y de no acotación 146
6.3.4 Fase de iteraciones estándar 147
6.3.5 El algoritmo MPE 149
6.3.6 Algunos ejemplos ilustrativos 150
Ejercicios 160

7 Programación lineal entera-mixta 163
7.1 Introducción 163
7.2 El método de ramificación y acotación 165
7.2.1 Introducción 165
7.2.2 El algoritmo de RA para un PPLEM 165
7.2.3 Estrategias de ramificación y procesamiento 166
7.2.4 Otros problemas de programación lineal entera-mixta . . 174
7.3 El método de los cortes de Gomory 175
7.3.1 Introducción 175
7.3.2 Generación de un corte 175
7.3.3 Algoritmo de los cortes de Gomory para PPLE 177
Ejercicios 183

8 Optimalidad y dualidad en programación no lineal 187
8.1 Introducción 187
8.2 Condiciones necesarias de optimalidad 193
8.2.1 Diferenciabilidad 193
8.2.2 Condiciones de Karush–Kuhn–Tucker 194
8.3 Condiciones de optimalidad: suficiencia y convexidad 211
8.3.1 Convexidad 211
8.3.2 Condiciones suficientes de Karush–Kuhn–Tucker 216
8.4 Teoría de la dualidad 222
8.5 Ilustración práctica de la dualidad y separabilidad 227
8.5.1 Esquema centralizado o método primal 227
8.5.2 Mercado competitivo o esquema dual 231
8.5.3 Conclusión 231
8.6 Condiciones de regularidad 232
Ejercicios 233

9 Métodos computacionales para programación no lineal 239
9.1 Algoritmos de optimización para problemas sin restricciones . . . 240
9.1.1 Métodos de búsqueda lineal 240
9.1.2 Optimización sin restricciones 246
9.2 Algoritmos de optimización con restricciones 259
9.2.1 Métodos duales 259
9.2.2 Métodos de penalizaciones 266
9.2.3 Método del punto interior en programación lineal 274
Ejercicios 283

III Software 289

10 La herramienta GAMS 291
10.1 Introducción 291
10.2 Ejemplo ilustrativo 292
10.3 Características del lenguaje 297
10.3.1 Conjuntos 298
10.3.2 Escalares 300
10.3.3 Vectores y matrices de datos 300
10.3.4 Reglas sobre las expresiones matemáticas en asignaciones 302
10.3.5 Variables 303
10.3.6 Restricciones 306
10.3.7 Modelos 306
10.3.8 Resolución 307
10.3.9 La potencia del asterisco 310
10.3.10 Resultados 311
10.3.11 Expresiones condicionales 311
10.3.12 Conjuntos dinámicos 313
10.3.13 Estructuras iterativas 313
10.3.14 Escritura en fichero de salida 316
10.3.15 Listado de las restricciones no lineales 317

11 Algunos ejemplos en GAMS 319
11.1 Introducción 319
11.2 Ejemplos de programación lineal 319
11.2.1 El problema del transporte 319
11.2.2 El problema de planificación de la producción 323
11.2.3 El problema de la dieta 325
11.2.4 El problema de flujos en redes 327
11.2.5 El problema de la cartera de valores 332
11.2.6 El sistema de vigas y cuerdas 334
11.2.7 El despacho económico de centrales térmicas 336
11.3 Ejemplos de programación lineal entera mixta 339
11.3.1 El problema de la mochila 339
11.3.2 La identificación de síntomas relevantes 341

11.3.3 El problema de la academia de ingeniería 342
11.3.4 El problema del horario 345
11.3.5 Modelos de localización de plantas productivas 347
11.3.6 Programación horaria de centrales térmicas 349
11.4 Ejemplos de programación no lineal 353
11.4.1 El ejemplo del paquete postal 353
11.4.2 El ejemplo de la tienda 354
11.4.3 El ejemplo de la lámpara 355
11.4.4 El ejemplo de la superficie 355
11.4.5 El ejemplo del transporte de arena 356
11.4.6 El ejemplo del voladizo 357
11.4.7 El ejemplo de la estructura con dos barras 357
11.4.8 El ejemplo de la columna 358
11.4.9 El ejemplo del sistema de vigas y cuerdas 359
11.4.10 Estimación de estado en sistemas eléctricos 361
11.4.11 Reparto óptimo de cargas 364
11.4.12 El problema de la red de abastecimiento de agua 367
11.4.13 El problema de la matriz equilibrada 369
11.4.14 El problema de la asignación de tráfico 371
Ejercicios 372

IV Aplicaciones 377

12 Aplicaciones 379
12.1 Aplicaciones a la inteligencia artificial 379
12.1.1 Aprendizaje de funciones neuronales 381
12.2 Aplicaciones a CAD 388
12.2.1 Generación automática de mallas 389
12.3 Aplicaciones a la probabilidad 395
12.3.1 Compatibilidad de matrices de probabilidad condicional . 396
12.3.2 Cuasi-compatibilidad 400
12.4 Modelos de regresión 404
12.5 Aplicaciones a problemas de optimización 410
12.5.1 Problemas de cálculo variacional 412
12.5.2 Problemas de control óptimo 419
12.6 Sistemas de transporte 426
12.6.1 Introducción 426
12.6.2 Elementos de una red de tráfico 428
12.6.3 El problema de asignación de tráfico 432
12.6.4 Modelos de asignación con restricciones laterales 440
12.6.5 El caso de la demanda elástica 443
12.6.6 Combinación de asignación y distribución 449
12.7 Coordinación hidrotérmica a corto plazo 453
12.7.1 Formulación del problema mediante RL 454
12.7.2 Resolución del problema dual 458

12.7.3 Significado económico de los multiplicadores 459

13 Algunos trucos útiles 461

13.1 Introducción 461
13.2 Algunos trucos genéricos 461
13.2.1 Tratamiento de variables no acotadas 462
13.2.2 Transformación de desigualdades en igualdades 463
13.2.3 Transformación de igualdades en desigualdades 464
13.2.4 Transformación de maximización a minimización 465
13.2.5 Transformación de funciones no lineales en lineales 465
13.2.6 Tratamiento de funciones no lineales como lineales 465
13.2.7 Espacio vectorial como cono 468
13.2.8 Restricciones alternativas 470
13.2.9 Tratamiento de restricciones condicionales 472
13.2.10 Tratamiento de funciones no continuas 472
13.2.11 Tratamiento de funciones no convexas a trozos 473
13.3 Algunos trucos en GAMS 476
13.3.1 Asignación de valores a una matriz 476
13.3.2 Definición de una matriz simétrica 477
13.3.3 Definición de una matriz cuasi-vacía 478
13.3.4 Descomposición de un problema separable 479
13.3.5 Adición iterativa de restricciones a un problema 480
13.3.6 Tratamiento de los estados inicial y final 481
13.3.7 Análisis de sensibilidad 482
13.3.8 Dependencia del flujo del programa 482
Ejercicios 483

A Soluciones factibles y compatibilidad 489

A.1 El cono dual 490
A.2 Cono asociado a un poliedro 492
A.3 El procedimiento **Γ** 495
A.4 Compatibilidad de sistemas lineales 501
A.5 Resolución de sistemas lineales 502
A.6 Aplicaciones a varios ejemplos 506
A.6.1 El problema del transporte 507
A.6.2 El problema de la planificación de la producción 512
A.6.3 Las tablas “input–output” 518
A.6.4 El problema de la dieta 520
A.6.5 El problema de las redes 521
Ejercicios 526

B Notación 531

Bibliografía 547

Índice 555

Prefacio

La modelización es una de las áreas más atractivas de la ingeniería y las ciencias aplicadas. De hecho, los ingenieros necesitan construir modelos para resolver problemas de la vida real. El objetivo de un modelo consiste en reproducir la realidad de la forma más fiel posible, tratando de entender cómo se comporta el mundo real y obteniendo las respuestas que pueden esperarse de determinadas acciones. En la práctica se utilizan muchos tipos de modelos, tales como modelos de ecuaciones diferenciales, modelos de ecuaciones funcionales, modelos en diferencias y de elementos finitos, y modelos de programación matemática.

La selección del modelo adecuado para reproducir la realidad es una etapa crucial para obtener una solución satisfactoria a un problema real. Las estructuras matemáticas asociadas no son arbitrarias, sino una consecuencia de la realidad misma. En este libro, se hace un esfuerzo importante por conectar las realidades física y matemática. Se muestra al lector el razonamiento que conduce al análisis de las diferentes estructuras, modelos y conceptos. Esto se pone de manifiesto en los ejemplos ilustrativos, que muestran la conexión entre modelo y realidad.

En este libro se tratan los modelos de programación matemática, incluyendo los de programación lineal y no lineal.

Los problemas de programación matemática son problemas particulares a los que uno se enfrenta con cierta frecuencia. Uno está preparado para resolverlos usando muchas de las herramientas disponibles, procedimientos o paquetes de software. De hecho, estos problemas se estudian en detalle en los estudios de grado y postgrado. Sin embargo, uno puede no estar preparado para resolver otros problemas muy frecuentes como

1. Problemas de programación lineal con muchas variables y/o restricciones

2. Problemas de programación no lineal

3. Técnicas de descomposición para problemas a resolver con herramientas de programación matemática

4. Reglas para transformar otros problemas en problemas de programación matemática

En este libro se dan métodos que permiten resolver una amplia colección de problemas prácticos interesantes.

El objetivo de este libro no es el de tratar los métodos estándar de análisis, que están ya cubiertos en muchos otros libros. Al contrario, se trata de mostrar al lector la potencia de estos métodos para resolver problemas prácticos, y luego de introducir los principales conceptos y herramientas.

Cuando se analiza y discute la programación matemática, una posibilidad es la de dar un profundo análisis teórico del problema y una discusión de los diferentes problemas y métodos. Esta opción tiene algunos riesgos. Aunque a veces, inicialmente, el tratamiento parece ser más riguroso y profundo, el lector es conducido a ser demasiado curioso y cuidadoso con los detalles matemáticos pero sin preocuparse ni entender a dónde conducen éstos o de dónde proceden. Por ejemplo, no es infrecuente dar a una persona que ha estudiado durante años programación lineal, un dibujo bidimensional sencillo en el que aparece el conjunto factible, y preguntarle que marque la secuencia de puntos extremos asociada al método simplex, sin obtener una respuesta correcta. Nótese que esto significa que no se comprende la esencia misma del método simplex y de las ideas en que éste se basa.

Alternativamente, uno puede tratar este tema con la ayuda de ejemplos ilustrativos, y tratar de transmitir al lector la profundidad y el ingenio que hay detrás de estos métodos, con lo que se hace el libro más legible y atractivo. No tratamos con métodos o soluciones estándar. El lector que busque métodos estándar o referencias de trabajos con esta orientación debería consultar uno de los muchos libros sobre este tema que se dan en la bibliografía. Por el contrario, en este libro se discuten los problemas antes mencionados desde otro punto de vista.

Además de obtener soluciones, matemáticos e ingenieros están interesados en analizar las condiciones que conducen a problemas bien definidos. En este contexto, los problemas de compatibilidad y unicidad de solución juegan un papel central. De hecho, conducen a conclusiones físicas e ingenieriles muy interesantes, que relacionan las condiciones físicas, obtenidas de la realidad, con las correspondientes condiciones que hay tras los modelos matemáticos. Los métodos a desarrollar en este libro también permiten concluir si el conjunto de restricciones conducen a la existencia de al menos una solución.

El libro está organizado en cuatro partes. En la primera parte se trata de los modelos para introducir al lector en el atractivo mundo de la programación matemática, por medio de ejemplos cuidadosamente seleccionados. Se guía al lector para que descubra el planteamiento matemático de los problemas tras un entendimiento claro de los problemas físicos e ingenieriles.

La parte segunda trata de los métodos y describe las técnicas principales para resolver problemas de programación lineal y no lineal. Tal como se ha indicado, la intención de esta parte no es la de dar un análisis riguroso de los métodos, sino un entendimiento de las ideas básicas y principales.

Puesto que nada puede hacerse en la práctica sin un software adecuado, en la parte tercera se ha seleccionado el GAMS (sistema general de modelización algebraica) como herramienta principal para ser usada en el libro. Para facilitar su uso, se da una descripción general de cómo deben plantearse los problemas y de las posibilidades de GAMS. Todos los problemas descritos en la primera

parte se resuelven usando esta herramienta.

Una vez que el lector está familiarizado con los problemas de programación matemática y el software GAMS, se dedica la parte cuarta a aplicaciones de estas técnicas a problemas prácticos más importantes de varias áreas del conocimiento, como la inteligencia artificial (AI), diseño asistido por ordenador (CAD), estadística y probabilidad, economía, ingeniería, y problemas de transporte. Todas estas aplicaciones han surgido de la vida real. En otras palabras, son modelos matemáticos de problemas físicos, económicos o ingenieriles. Esta parte incluye también un capítulo dedicado a trucos que son útiles cuando se tratan casos especiales de problemas, como, por ejemplo, convertir problemas no lineales en lineales, y también se dan algunos trucos específicos de GAMS.

Este libro puede utilizarse como libro de consulta o referencia y como libro de texto en cursos de grado y de postgrado. Se incluyen en el libro numerosos ejemplos ilustrativos y ejercicios de fin de capítulo. Además se ha incluido el código GAMS para implementar muchos de los ejemplos del libro. La versión actual de este programa, junto con una breve *Guía de usuario*, puede ser obtenida por Internet en la dirección:

`http://www.gams.com/`

Este libro está dirigido a una audiencia amplia, que incluye matemáticos, ingenieros, y científicos aplicados. Hay unos pocos prerrequisitos para el lector, ya que un conocimiento previo de álgebra lineal, cálculo elemental y una cierta familiaridad con matrices son muy convenientes.

Varios colegas y estudiantes han leído versiones previas de los manuscritos de este libro y han suministrado valiosos comentarios y sugerencias. Sus contribuciones han dado lugar a una sustancial mejora del libro. En particular, se agradece la ayuda de Angel Yustres, José Manuel Vellón, y Marta Serrano, que hicieron una lectura muy cuidadosa del manuscrito y dieron interesantes sugerencias, a Gonzalo Echevarría, por su ayuda informática, y al equipo de Wiley por su trabajo en la edición en versión inglesa de este libro.

Los autores agradecen también a la Universidad de Cantabria, e Iberdrola y José Antonio Garrido por su soporte económico. Especial mención merece la Universidad de Castilla-La Mancha por la publicación y financiación de este trabajo.

Finalmente, Enrique Castillo quiere expresar su agradecimiento más sincero al Profesor Alfonso Fernández Canteli (Universidad de Oviedo) por su constante apoyo, ánimo y confianza.

Enrique Castillo, Antonio J. Conejo,
Pablo Pedregal, Ricardo García,
y Natalia Alguacil

Ciudad Real, España
20 de febrero de 2002

Parte I

Modelos

Capítulo 1

Programación lineal

1.1 Introducción

La programación matemática es una potente técnica de modelado usada en el proceso de toma de decisiones. Cuando se trata de resolver un problema de este tipo, la primera etapa consiste en identificar las posibles decisiones que pueden tomarse; esto lleva a identificar las variables del problema concreto. Normalmente, las variables son de carácter cuantitativo y se buscan los valores que optimizan el objetivo. La segunda etapa supone determinar qué decisiones resultan admisibles; esto conduce a un conjunto de restricciones que se determinan teniendo presente la naturaleza del problema en cuestión. En la tercera etapa, se calcula el coste/beneficio asociado a cada decisión admisible; esto supone determinar una función objetivo que asigna, a cada conjunto posible de valores para las variables que determinan una decisión, un valor de coste/beneficio. El conjunto de todos estos elementos define el problema de optimización.

La programación lineal (PL), que trata exclusivamente con funciones objetivos y restricciones lineales, es una parte de la programación matemática, y una de las áreas más importantes de la matemática aplicada. Se utiliza en campos como la ingeniería, la economía, la gestión, y muchas otras áreas de la ciencia, la técnica y la industria.

En este capítulo se introduce la programación lineal por medio de varios ejemplos seleccionados. Para empezar nuestra exposición se hace notar que cualquier problema de programación lineal requiere identificar cuatro componentes básicos:

1. El conjunto de datos.

2. El conjunto de variables involucradas en el problema, junto con sus dominios respectivos de definición.

3. El conjunto de restricciones lineales del problema que definen el conjunto de soluciones admisibles.

4. La función lineal que debe ser optimizada (minimizada o maximizada).

En las secciones que siguen se da una lista de ejemplos, prestando especial atención en cada caso a estos cuatro elementos.

La lista seleccionada no es sino una muestra de la gran cantidad de problemas de programación lineal (PPL) disponibles en las referencias. El objetivo en dicha selección es poder ilustrar de manera clara el alcance de la programación lineal y ayudar a nuestros lectores a familiarizarse con los cuatro elementos descritos más arriba.

1.2 El problema del transporte

En esta sección se presenta y se describe el problema del transporte.

Imagínese que un cierto producto debe enviarse en determinadas cantidades $u_1, \ldots, u_m$, desde cada uno de m orígenes, y recibirse en cantidades $v_1, \ldots, v_n$, en cada uno de n destinos. El problema consiste en determinar las cantidades x_{ij}, que deben enviarse desde el origen i al destino j, para conseguir minimizar el coste del envío.

Los cuatro elementos principales de este problema son:

1. **Datos**

   m: el número de orígenes

   n: el número de destinos

   u_i: la cantidad que debe enviarse desde el origen i

   v_j: la cantidad que debe ser recibida en el destino j

   c_{ij}: el coste de envío de una unidad de producto desde el origen i al destino j

2. **Variables**

   x_{ij}: la cantidad que se envía desde el origen i al destino j.
   Se supone que las variables deben ser no negativas:

$$x_{ij} \geq 0;\ i = 1, \ldots, m;\ j = 1, \ldots, n \tag{1.1}$$

   Esto implica que la dirección de envío del producto está prefijada desde los distintos orígenes hasta los destinos. No obstante, otras hipótesis podrían tenerse en cuenta. Por ejemplo, podría no limitarse el signo de las variables $x_{ij} \in \mathbb{R}$, si no se quiere predeterminar cuáles son los puntos de partida y llegada.

3. **Restricciones.** Las restricciones de este problema son:

$$\begin{array}{rcl} \sum\limits_{j=1}^{n} x_{ij} & = & u_i;\ i = 1, \ldots, m \\ \sum\limits_{i=1}^{m} x_{ij} & = & v_j;\ j = 1, \ldots, n \end{array} \tag{1.2}$$

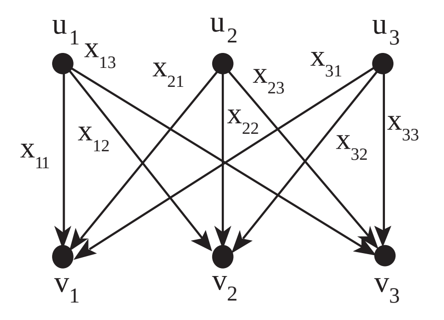


Figura 1.1: Esquema del problema del transporte.

El primer conjunto de condiciones indica que la cantidad del producto que parte del origen i debe coincidir con la suma de las cantidades que parten de ese origen hasta los distintos destinos $j = 1, \ldots, n$.

El segundo conjunto de condiciones asegura que el total recibido en el destino j debe corresponder a la suma de todas las cantidades que llegan a ese destino y parten de los distintos orígenes $i = 1, \ldots, m$.

Nuestros lectores deben distinguir entre las *cotas de las variables* (1.1) y las *restricciones del problema* (1.2).

4. **Objetivo que debe optimizarse.** En el problema del transporte nos interesa normalmente minimizar los costes de envío (suma de los costes de envío por unidad de producto multiplicado por las cantidades enviadas); es decir, se debe minimizar

$$Z = \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} \tag{1.3}$$

Una vez que se han identificado estos cuatro elementos, se está preparado para resolver el problema.

Ejemplo 1.1 (problema del transporte). Considérese el problema del transporte de la Figura 1.1, con $m = 3$ orígenes y $n = 3$ destinos, y

$$u_1 = 2, u_2 = 3, u_3 = 4; v_1 = 5, v_2 = 2, v_3 = 2$$

En este caso, el sistema (1.2) es

$$
\mathbf{C}\mathbf{x} = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_{11} \\ x_{12} \\ x_{13} \\ x_{21} \\ x_{22} \\ x_{23} \\ x_{31} \\ x_{32} \\ x_{33} \end{pmatrix} = \begin{pmatrix} 2 \\ 3 \\ 4 \\ 5 \\ 2 \\ 2 \end{pmatrix} \tag{1.4}
$$

$$
x_{ij} \geq 0; \quad i, j = 1, 2, 3
$$

Las tres primeras ecuaciones establecen la conservación del producto en los tres orígenes y las tres últimas igualdades, la conservación del producto en los tres destinos.

Si se concretan valores particulares

$$
\mathbf{c} = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 1 & 2 \\ 3 & 2 & 1 \end{pmatrix}
$$

para los costes de envío, nuestro problema consiste en minimizar

$$
Z = x_{11} + 2x_{12} + 3x_{13} + 2x_{21} + x_{22} + 2x_{23} + 3x_{31} + 2x_{32} + x_{33}. \tag{1.5}
$$

Mediante el paquete GAMS puede resolverse este problema (véase la sección 11.2.1), y encontrar un valor mínimo de 14 para la función objetivo, lo que implica un coste de

$$
Z = 14 \text{ unidades, para } \mathbf{x} = \begin{pmatrix} 2 & 0 & 0 \\ 1 & 2 & 0 \\ 2 & 0 & 2 \end{pmatrix} \tag{1.6}
$$

Típicamente, los paquetes comerciales de optimización proporcionan soluciones óptimas o informan al usuario de que el problema es no factible, pero no indican si existen diversas soluciones óptimas. En la sección A.40 del apéndice, se verá que este problema admite muchas soluciones óptimas con el mismo coste óptimo asociado.

■

1.3 El problema de la planificación de la producción

Un productor fabrica una pieza, cuya demanda varía en el tiempo, de acuerdo con el gráfico de la Figura 1.2. El productor debe siempre atender la demanda

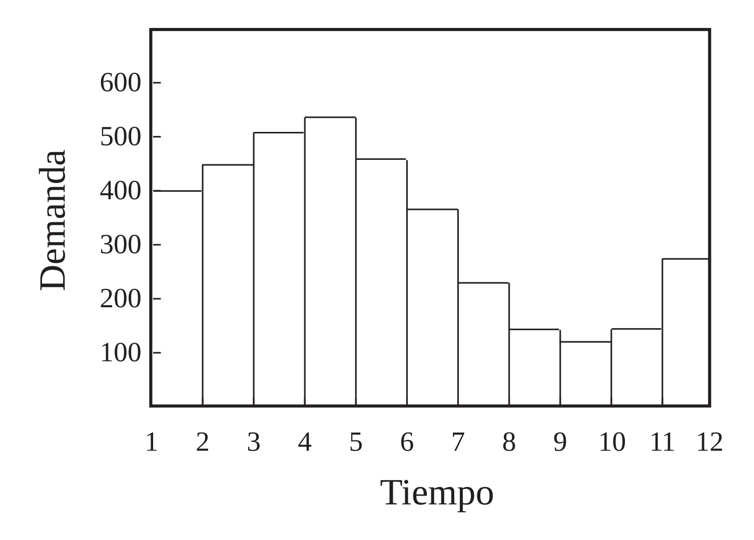


Figura 1.2: Gráfico de la demanda en función del tiempo.

mensual. En general, cualquier problema de planificación admitirá diversas posibilidades que aseguren que la demanda es convenientemente atendida. Existen dos posibilidades:

1. **Producción variable**. El fabricante puede producir cada mes el número exacto de unidades que le solicitan. Sin embargo, como una producción que varía es costosa de mantener, por los costes de horarios más largos en los meses de producción alta y los costes asociados al paro del personal y la maquinaria en los meses de producción baja; este tipo de producción no es eficiente.

2. **Producción constante**. El fabricante que debe atender una demanda que cambia con el tiempo puede producir por encima de dicho nivel en periodos de baja demanda y almacenar la sobreproducción para los periodos de demanda mayor. Así, la producción puede mantenerse constante, compensando la demanda alta con la sobreproducción de periodos pasados. Sin embargo, debido a los costes de almacenamiento, tal opción puede no ser deseable si requiere costes altos de almacenamiento durante varios meses.

Los problemas de esta naturaleza ilustran las dificultades que surgen cuando objetivos contrarios están presentes en un sistema dado. Nuestro objetivo es llevar a cabo una planificación de la producción que maximice los beneficios después de considerar los costes de las variaciones en la producción y los almacenes.

Los cuatro elementos principales que intervienen en el problema de la planificación de la producción son:

1. **Datos**

n: el número de meses a considerar

s_0: la cantidad almacenada disponible al principio del periodo considerado

d_t: el número de unidades (demanda) que se solicita en el mes t

s^{max}: la capacidad máxima de almacenamiento

a_t: el precio de venta en el mes t

b_t: el coste de producción en el mes t

c_t: el coste de almacenamiento en el mes t

2. **Variables**

x_t: el número de unidades producidas en el mes t

s_t: el número de unidades almacenadas en el mes t

3. **Restricciones.** Como la demanda d_t en el mes t debe coincidir con el cambio en el almacenamiento, $s_{t-1} - s_t$, más la producción x_t en el mes t; la capacidad de almacenamiento no puede excederse; y la demanda d_t, almacenamiento s_t, y producción x_t deben ser no negativas; se tienen las siguientes restricciones:

$$\begin{array}{lcl} s_{t-1} + x_t - d_t & = & s_t;\ \ t = 1, 2, \ldots, n \\ s_t & \leq & s^{max};\ \ t = 1, 2, \ldots, n \\ s_t, x_t & \geq & 0 \end{array} \tag{1.7}$$

4. **Función a optimizar.** Una posibilidad en el problema de la planificación de la producción consiste en maximizar el ingreso después de descontar los costes de la variación de la producción y los inventarios; esto es, maximizar el beneficio

$$Z = \sum_{t=1}^{n} (a_t d_t - b_t x_t - c_t s_t) \tag{1.8}$$

Si el periodo es corto, a_t, b_t, y c_t pueden considerarse constantes, esto es, $a_t = a, b_t = b$ y $c_t = c$.

Otra posibilidad consiste en minimizar los costes de almacenamiento:

$$Z = \sum_{t=1}^{n} c_t s_t \tag{1.9}$$

Ejemplo 1.2 (problema de la planificación de la producción). En este ejemplo se ilustra el problema de la planificación de la producción sin límite en la capacidad de almacenamiento. Considérese la función de demanda en la tabla

Tabla 1.1: Demanda como función del tiempo en el ejemplo 1.2

Tiempo	Demanda
1	2
2	3
3	6
4	1

1.1 y supóngase que la cantidad almacenada inicialmente es $s_0 = 2$. Entonces, el sistema (1.7) se transforma en

$$\mathbf{Cx} = \begin{pmatrix} -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} s_1 \\ s_2 \\ s_3 \\ s_4 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 0 \\ 3 \\ 6 \\ 1 \end{pmatrix}$$

$$s_t, x_t \geq 0; \quad t = 1, 2, 3, 4 \tag{1.10}$$

donde el 0 en la matriz de la derecha procede de restar la demanda para $t = 1$ del almacenamiento inicial.

Si se maximiza el beneficio después de descontar los costes debidos a las variaciones en la producción y los inventarios, como en (1.8), y se toma $a_t = 3, b_t = 1, c_t = 1$, nuestro problema de optimización se traduce en maximizar

$$Z = 36 - x_1 - x_2 - x_3 - x_4 - s_1 - s_2 - s_3 - s_4 \tag{1.11}$$

sujeto a (1.10).

Mediante GAMS puede resolverse este problema (véase la sección 11.2.2), y obtener el valor máximo

$$Z = 26 \text{ para } \mathbf{x} = (s_1, s_2, s_3, s_4, x_1, x_2, x_3, x_4) = (0, 0, 0, 0, 0, 3, 6, 1)^T$$

lo que implica ningún almacenamiento. Se sugiere al lector que concrete una nueva función de demanda de modo que la solución óptima de lugar a almacenamiento no nulo.

1.4 El problema de la dieta

El problema de la dieta consiste en determinar las cantidades de distintos nutrientes que deben ingerirse asegurar ciertas condiciones de nutrición y minimizar el coste de compra de los nutrientes. De modo más preciso, supóngase que se

conocen los contenidos nutritivos de ciertos alimentos, sus precios y la cantidad mínima diaria de nutrientes aconsejada. El problema consiste en determinar la cantidad de cada alimento que debe comprarse de suerte que se satisfagan los mínimos aconsejados y se alcance un precio total mínimo.

Los cuatro elementos que intervienen en el problema de la dieta son:

1. **Datos**

   m: el número de nutrientes

   n: el número de alimentos

   a_{ij}: la cantidad del nutriente i en una unidad del alimento j

   b_i: la cantidad mínima del nutriente i aconsejada

   c_j: el precio de una unidad del alimento j

2. **Variables.** Las variables relevantes en este problema son:

   x_j: la cantidad del alimento j que debe adquirirse.

3. **Restricciones.** Como la cantidad total de un nutriente dado i es la suma de las cantidades de los nutrientes en todos los alimentos y las cantidades de alimentos deben ser no negativas, se deben cumplir las siguientes restricciones:

$$\begin{array}{rcl} \sum_{j=1}^{n} a_{ij}x_j & \geq & b_i; \;\; i = 1, \ldots, m \\ x_j & \geq & 0; \;\; j = 1, \ldots, n \end{array} \tag{1.12}$$

4. **Función a minimizar.** En el problema de la dieta se está interesado en minimizar el precio de la dieta:

$$Z = \sum_{j=1}^{n} c_j x_j \tag{1.13}$$

   donde c_j es el precio unitario del alimento j.

■

Ejemplo 1.3 (el problema de la dieta). Considérese un caso con cinco nutrientes y con los mínimos aconsejados para los nutrientes digeribles (DN), proteínas digeribles (DP), calcio (Ca), y fósforo (Ph) dados en la tabla 1.2.

Las restricciones (1.12) se convierten en

$$\begin{array}{c} \begin{pmatrix} 78.6 & 70.1 & 80.1 & 67.2 & 77.0 \\ 6.50 & 9.40 & 8.80 & 13.7 & 30.4 \\ 0.02 & 0.09 & 0.03 & 0.14 & 0.41 \\ 0.27 & 0.34 & 0.30 & 1.29 & 0.86 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{pmatrix} \geq \begin{pmatrix} 74.2 \\ 14.7 \\ 0.14 \\ 0.55 \end{pmatrix} \\ x_1, x_2, x_3, x_4, x_5 \geq 0 \end{array} \tag{1.14}$$

Tabla 1.2: Contenidos nutritivos de cinco alimentos: (DN) nutrientes digeribles, (DP) proteínas digeribles, (Ca) calcio, y (Ph) fósforo

Nutriente	Cantidad requerida	Maíz A	Avena	Maíz B	Salvado	Linaza
DN	74.2	78.6	70.1	80.1	67.2	77.0
DP	14.7	6.50	9.40	8.80	13.7	30.4
Ca	0.14	0.02	0.09	0.03	0.14	0.41
Ph	0.55	0.27	0.34	0.30	1.29	0.86

Supóngase que los precios unitarios de los alimentos son:

$$c_1 = 1,\ c_2 = 0.5,\ c_3 = 2,\ c_4 = 1.2, c_5 = 3.$$

De este modo, se tiene el PPL siguiente; minimizar

$$Z = x_1 + 0.5x_2 + 2x_3 + 1.2x_4 + 3x_5 \tag{1.15}$$

sujeto a (1.14).

Usando alguno de los paquetes existentes para resolver dicho PPL, como por ejemplo GAMS, se llega a la solución con un coste mínimo de (ver el código al final de la sección 11.6) $Z = 0.793$, en el punto $(0, 1.530, 0, 0.023, 0)$. Esto significa que sólo debe comprarse avena y salvado.

■

1.5 El problema del flujo en una red

Considérese una red de transporte (un sistema de tuberías, ferrocarriles, autopistas, comunicaciones, etc.) a través del cual desea mandarse un producto homogéneo (aceite, grano, coches, mensajes, etc.) desde ciertos puntos de la red, llamados *nudos fuente*, hasta otros nudos de destino, llamados *sumideros*. Además de estas dos clases de nudos, la red puede contener nudos intermedios, donde no se genera ni se consume el producto que está fluyendo por la red. Denótese por x_{ij} el flujo que va desde el nudo i al nudo j (positivo en la dirección $i \to j$, y negativo en la dirección contraria).

Los cuatro elementos presentes en los problemas de flujo son

1. **Datos**

   $\mathcal{G}$: el grafo $\mathcal{G} = (\mathcal{N}, \mathcal{A})$ que describe la red de transporte, donde $\mathcal{N}$ es el conjunto de nudos, y $\mathcal{A}$ es el conjunto de conexiones

   n: el número de nudos en la red

   f_i: el flujo entrante (positivo) o saliente (negativo) en el nudo i

   m_{ij}: la capacidad máxima de flujo en la conexión entre el nudo i y el j

c_{ij}: el precio de mandar una unidad del bien desde el nudo i al nudo j.

2. **Variables.** Las variables involucradas en este problema son:

   x_{ij}: el flujo que va desde el nudo i al nudo j.

3. **Restricciones.** Imponiendo la condición de conservación del flujo en todos los nudos, y las restricciones sobre la capacidad de las líneas o conexiones, se obtienen las siguientes restricciones. Las referidas a la conservación del flujo son

$$\sum_j (x_{ij} - x_{ji}) \quad = \quad f_i; \quad i = 1, \dots, n \tag{1.16}$$

   y las relacionadas con la capacidad de las líneas o conexiones son

$$-m_{ij} \quad \leq x_{ij} \leq \quad m_{ij}; \quad \forall i < j \tag{1.17}$$

   donde $i < j$ evita la posible duplicación de restricciones.

4. **Función a minimizar.** El precio total es

$$Z = \sum_{ij} c_{ij} x_{ij} \tag{1.18}$$

   Así, debe minimizarse (1.18) bajo (1.16) y (1.17).

Los problemas de flujo en redes son abundantes en ingeniería. De hecho, los sistemas de abastecimiento de agua, los sistemas de redes de comunicación, y otros, conducen a este tipo de problemas. Además de encontrar las soluciones de los problemas de optimización, se puede estar interesado en analizar el conjunto de soluciones, y en cómo cambia dicho conjunto cuando fallan algunos elementos en el sistema. El ejemplo siguiente se centra en una de tales situaciones.

Ejemplo 1.4 (un problema de flujo en redes). Considérese el problema de flujo en la red de la figura 1.3, donde las flechas indican los valores positivos de las variables del flujo.

Las ecuaciones (1.16) y (1.17) son ahora

$$\begin{pmatrix} 1 & 1 & 1 & 0 & 0 \\ -1 & 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 & 1 \\ 0 & 0 & -1 & -1 & -1 \end{pmatrix} \begin{pmatrix} x_{12} \\ x_{13} \\ x_{14} \\ x_{24} \\ x_{34} \end{pmatrix} = \begin{pmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{pmatrix} \tag{1.19}$$

$$\begin{aligned} x_{ij} &\leq m_{ij}; \ \forall i < j \\ -x_{ij} &\leq m_{ij}; \ \forall i < j \end{aligned}$$

donde se supone que $m_{ij} = 4, \forall i < j$, y $(f_1, f_2, f_3, f_4) = (7, -4, -1, -2)$.

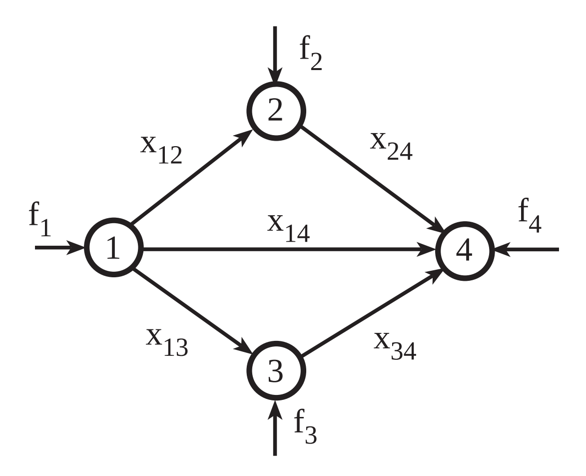


Figura 1.3: Diagrama de un problema de flujo en redes.

Supóngase además que $c_{ij} = 1; \forall i, j$. El problema de optimización es minimizar

$$Z = x_{12} + x_{13} + x_{14} + x_{24} + x_{34}$$

sujeto a (1.19).

Mediante el software adecuado, puede obtenerse la solución siguiente:

$$Z = 5 \text{ en el punto } \begin{pmatrix} x_{12} \\ x_{13} \\ x_{14} \\ x_{24} \\ x_{34} \end{pmatrix} = \lambda \begin{pmatrix} 0 \\ 3 \\ 4 \\ -4 \\ 2 \end{pmatrix} + (1-\lambda) \begin{pmatrix} 4 \\ -1 \\ 4 \\ 0 \\ -2 \end{pmatrix} ; \quad 0 \leq \lambda \leq 1$$

Esta solución indica que existe un conjunto infinito de soluciones, todas ellas proporcionando el mismo valor óptimo, $Z = 5$. En el capítulo 5, sección A.6.5, se explicará cómo se han obtenido todas estas soluciones óptimas.

■

1.6 El problema de la cartera de valores

Un inversor es propietario de participaciones de varios valores. Más concretamente, es dueño de b_i participaciones de los valores bursátiles A_i, $i = 1, 2, ..., m$. Los precios actuales de estos valores son v_i. Considérese que se pueden predecir los dividendos que se pagarán al final del año que comienza y los precios finales de los diferentes valores bursátiles, esto es, A_i pagará d_i y tendrá un nuevo precio w_i.

Nuestro objetivo es ajustar nuestra cartera, es decir, el número de participaciones en cada valor, de modo que se maximicen los dividendos. Nuestras incógnitas son x_i, el cambio en el número de participaciones que ya se tienen.

Como en un principio se tenían b_i participaciones del valor bursátil i, después del ajuste se tendrán $b_i + x_i$.

Los elementos principales del problema de la cartera son:

1. **Datos**

   m: el número de valores bursátiles

   b_i: el número actual de participaciones del valor bursátil i

   v_i: el precio actual del valor i por participación

   d_i: el dividendo que se pagará al final del año en el valor bursátil i

   w_i: el nuevo precio del valor bursátil i

   r: porcentaje mínimo r del valor actual de toda la cartera que no debe superarse en el ajuste

   s: porcentaje mínimo del valor total actual que no debe superarse por el valor futuro total de la cartera, para hacer frente a la inflación.

2. **Variables**

   x_i: el cambio en el número de participaciones del valor bursátil i.

3. **Restricciones.** Aunque no se ha indicado en el enunciado del problema, deben asegurarse ciertas condiciones que debe satisfacer una cartera bien equilibrada, como las restricciones que siguen a continuación. El número de participaciones debe ser no negativo:

   $$x_i \geq -b_i \tag{1.20}$$

   La cartera debe evitar depender en exceso de un valor cualquiera; esta condición puede establecerse exigiendo que el capital asociado a todo valor concreto, después del ajuste, represente al menos una cierta fracción r del capital total actual de la cartera. Esto es:

   $$r(\sum_i v_i(b_i + x_i)) \leq v_j(b_j + x_j); \quad \forall j \tag{1.21}$$

   El capital total de la cartera no debe cambiar en el ajuste pues se supone que no se invierte dinero adicional:

   $$\sum_i v_i x_i = 0 \tag{1.22}$$

   Para hacer frente a la inflación, el capital total en el futuro debe ser al menos un cierto porcentaje s mayor que el capital invertido actualmente:

   $$\sum_i w_i(b_i + x_i) \geq (1 + s) \sum_i v_i b_i \tag{1.23}$$

4. **Función a optimizar.** Nuestro objetivo es maximizar los dividendos:

$$Z = \sum_i d_i(b_i + x_i) \tag{1.24}$$

Nuestra tarea se concreta en determinar el valor máximo de los dividendos sujeto a todas las restricciones anteriores.

Ejemplo 1.5 (el problema de la cartera). Estúdiese el caso particular en que se tienen participaciones de tres valores bursátiles, 75 de A, 100 de B, y 35 de C, con precios 20, 20 y 100 dólares, respectivamente. Además se dispone de la siguiente información: A no pagará dividendos y alcanzará una nueva cotización de 18 dólares, B pagará 3 dólares por participación y la nueva cotización será 23 dólares, y C pagará 5 dólares por participación con una nueva cotización de 102 dólares. Si se toman los porcentajes r como 0.25 y s, 0.30, todas las restricciones anteriores se escriben

$$\begin{array}{rcl} x_A & \geq & -75 \\ x_B & \geq & -100 \\ x_C & \geq & -35 \\ 0.25\,[20(75+x_A)+20(100+x_B)+100(35+x_C)] & \leq & 20(75+x_A) \\ 0.25\,[20(75+x_A)+20(100+x_B)+100(35+x_C)] & \leq & 20(100+x_B) \\ 0.25\,[20(75+x_A)+20(100+x_B)+100(35+x_C)] & \leq & 100(35+x_C) \\ 20x_A+20x_B+100x_C & = & 0 \\ 18(75+x_A)+23(100+x_B)+102(35+x_C) & \geq & 1.03(7000) \end{array} \tag{1.25}$$

Después de varias simplificaciones, las restricciones anteriores se transforman en

$$\begin{array}{rcr} x_A & \geq & -75 \\ x_B & \geq & -100 \\ x_C & \geq & -35 \\ 15x_A - 5x_B - 25x_C & \geq & 250 \\ -5x_A + 15x_B - 25x_C & \geq & -250 \\ -5x_A - 5x_B + 75x_C & \geq & -1750 \\ 20x_A + 20x_B + 100x_C & = & 0 \\ 18x_A + 23x_B + 102x_C & \geq & 270 \end{array} \tag{1.26}$$

La solución que proporciona GAMS es

$$Z = 612.5 \text{ dólares y } x_A = 12.5, \quad x_B = 75, \quad x_C = -17.5$$

■

1.7 El sistema de vigas y cuerdas

Este sistema consta de varias cuerdas y vigas conectadas de un modo particular. Varias cargas externas actúan en el punto medio de algunas vigas. El problema

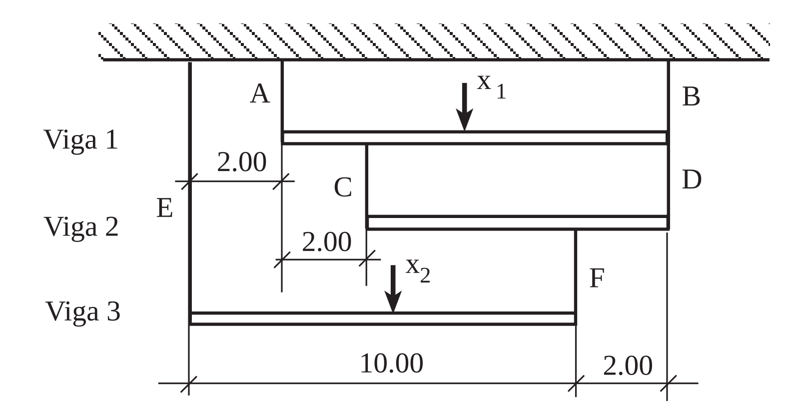


Figura 1.4: Sistema de cuerdas y vigas.

consiste en determinar la carga total admisible que puede soportar tal sistema sin colapsar, bajo equilibrio de fuerzas y de momentos, si se supone que el peso de las cuerdas y las vigas es despreciable.

Los principales elementos que intervienen en este problema son

1. **Datos**

   I: conjunto de cargas

   S: conjunto de cuerdas

   B: conjunto de vigas

   T_s: carga máxima permitida en la cuerda $s \in S$

   Ω_b: conjunto de cargas aplicadas en el punto medio de la viga b. Obsérvese que $\Omega_b \subset I$ y consiste en una sola carga, o ninguna

   Ψ_b: conjunto de cuerdas que soportan la viga b; Ψ_b es un subconjunto de S y normalmente consta de dos elementos

   Θ_b: conjunto de cuerdas que cuelgan de la viga b

   dl_i: distancia de la carga i al punto izquierdo de la viga donde actúa

   dr_s: distancia de la cuerda s al punto izquierdo de la viga b que soporta. $s \in \Psi_b$.

2. **Variables.** Las variables involucradas en este problema son las siguientes:

   x_i: la carga i

   t_s: tensión generada en la cuerda s bajo la acción del conjunto de las cargas x_i, $i \in I$.

3. **Restricciones.** La condición de equilibrio de fuerzas en cada viga lleva al conjunto de ecuaciones

$$\sum_{s \in \Psi_b} t_s = \sum_{i \in \Omega_b} x_i + \sum_{x \in \Theta_b} t_s$$

para cada $b \in B$, y la condición de equilibrio de momentos (tomados con respecto a los extremos izquierdos de cada viga) se escribe como

$$\sum_{s \in \Psi_b} dr_s t_s = \sum_{i \in \Omega_b} dl_i x_i + \sum_{x \in \Theta_b} dr_s t_s$$

para cada viga $b \in B$.

También se debe respetar la tensión máxima permitida en cada cuerda

$$0 \leq t_s \leq T_s$$

para cada $s \in S$; y la no negatividad de cada carga i

$$x_i \geq 0$$

4. **Función a optimizar.** El problema consiste en maximizar la carga total:

$$Z = \sum_i x_i$$

Ejemplo 1.6 (un sistema de vigas y cuerdas). Como ejemplo concreto considérese el sistema descrito en la figura 1.4, donde las cargas x_1 y x_2 se aplican en los puntos medios de las vigas 2 y 3, respectivamente. Las cuerdas A y B pueden soportar una carga máxima de 300; C y D, 200; y E y F, 100.

La condición de equilibrio de fuerzas en cada viga se plasma en las ecuaciones

$$\begin{array}{rcl} t_E + t_F & = & x_2 \\ t_C + t_D & = & t_F \\ t_A + t_B & = & x_1 + t_C + t_D \end{array} \tag{1.27}$$

y el equilibrio de momentos (tomados con respecto a E, C, y A, respectivamente) proporciona

$$\begin{array}{rcl} 10t_F & = & 5x_2 \\ 8t_D & = & 6t_F \\ 10t_B & = & 5x_1 + 2t_C + 10t_D \end{array} \tag{1.28}$$

Resolviendo sucesivamente estas ecuaciones para las tensiones, se llega a

$$\begin{array}{rcl} t_F & = & \dfrac{x_2}{2} \\ t_E & = & \dfrac{x_2}{2} \\ t_D & = & \dfrac{3x_2}{8} \\ t_C & = & \dfrac{x_2}{8} \\ t_B & = & \dfrac{2x_2}{5} + \dfrac{x_1}{2} \\ t_A & = & \dfrac{x_2}{10} + \dfrac{x_1}{2} \end{array} \tag{1.29}$$

Finalmente, considerando que las cargas actúan hacia abajo ($x_1, x_2 \geq 0$), las restricciones anteriores se transforman en:

$$\begin{array}{rcl} \dfrac{x_2}{2} & \leq & 100 \\ \dfrac{3x_2}{8} & \leq & 200 \\ \dfrac{x_2}{8} & \leq & 200, \\ \dfrac{x_1}{2} + \dfrac{2x_2}{5} & \leq & 300 \\ \dfrac{x_1}{2} + \dfrac{x_2}{10} & \leq & 300 \\ x_1 & \geq & 0 \\ x_2 & \geq & 0 \end{array} \tag{1.30}$$

En realidad, de entre todas estas condiciones hay algunas que son más restrictivas que otras. De hecho, las restricciones anteriores pueden reducirse al conjunto de restricciones:

$$\begin{array}{rcl} 0 \ \leq \ x_2 & \leq & 200 \\ 4x_2 + 5x_1 & \leq & 3000 \\ x_1 & \geq & 0 \end{array} \tag{1.31}$$

El problema consiste en maximizar la carga total

$$x_1 + x_2$$

sujeto a las restricciones (1.31).

La solución que aporta GAMS, descrita en la sección 11.2.6, es

$$Z = 640 \text{ en el punto } x_1 = 440; \ \ x_2 = 200$$

Las tensiones correspondientes en las cuerdas son

$$t_A = 240; \ \ t_B = 300; \ \ t_C = 25; \ \ t_D = 75; \ \ t_E = 100; \ \ t_F = 100$$

■

1.8 El problema del despacho económico

Los generadores de energía eléctrica así como las demandas están localizados en distintos nudos de una red eléctrica. El objetivo del problema del despacho económico es calcular, para un instante determinado, la potencia que ha de producir cada generador de modo que se satisfaga la demanda a un coste mínimo, al tiempo que se cumplan distintas restricciones técnicas de la red y los generadores.

Cada línea en la red eléctrica transmite la potencia desde el nudo suministrador hasta el nudo receptor. La cantidad de potencia enviada es proporcional a la diferencia de los ángulos de estos nudos (al igual que el flujo de agua a través de la tubería que conecta dos depósitos es proporcional a la diferencia de alturas de las superficies en los dos depósitos). La constante de proporcionalidad tiene un nombre curioso pues los ingenieros eléctricos la denominan *susceptancia*. La potencia transmitida desde el nudo i al nudo j a través de la línea $i-j$ es por tanto

$$B_{ij}(\delta_i - \delta_j) \tag{1.32}$$

donde B_{ij} es la susceptancia de la línea $i-j$; y δ_i y δ_j los ángulos de los nudos i y j, respectivamente.[1]

Por razones físicas, la cantidad de potencia transmitida a través de una línea de potencia tiene un límite. Dicho límite se justifica mediante consideraciones térmicas o de estabilidad. Por tanto, una línea debe funcionar de manera que este límite de transporte no se supere en ningún caso. Esto se formula como

$$-P_{ij}^{max} \leq B_{ij}(\delta_i - \delta_j) \leq P_{ij}^{max} \tag{1.33}$$

donde P_{ij}^{max} es la capacidad máxima de transporte de la línea $i-j$.

Debe insistirse en que la potencia transmitida es proporcional a la diferencia de ángulos y no a un ángulo dado. En consecuencia, el valor de un ángulo arbitrario puede fijarse a 0 y considerarlo como el origen:

$$\delta_k = 0 \tag{1.34}$$

donde k es un nudo arbitrario.

Una consecuencia de seleccionar de modo arbitrario el origen es que los ángulos son variables no restringidas en signo.

La potencia producida por un generador es una magnitud positiva acotada inferior y superiormente. La cota inferior se debe a condiciones de estabilidad (de manera análoga a como un coche no puede moverse con velocidades por debajo de un cierto límite). La cota superior obedece a consideraciones térmicas (al igual que la velocidad de un vehículo no puede superar una cierta cota superior). Las restricciones anteriores pueden expresarse como

$$P_i^{min} \leq p_i \leq P_i^{max} \tag{1.35}$$

donde p_i es la potencia producida por el generador i y P_i^{min} y P_i^{max} son constantes positivas representando, respectivamente, la potencia de salida máxima y mínima admisibles para el generador i.

En cada nudo, la potencia que llega debe coincidir con la potencia que sale del mismo (ley de conservación de energía), que puede expresarse como

$$\sum_{j \in \Omega_i} B_{ij}(\delta_j - \delta_i) + p_i = D_i, \ \ \forall i \tag{1.36}$$

[1]Existen modelos más precisos, pero el que se presenta aquí es razonable para muchos análisis técnicos y económicos.

donde Ω_i es el conjunto de nudos conectados a través de las líneas al nudo i y D_i la demanda en el nudo i.

Según se ha indicado antes, la potencia transmitida a través de cada línea está acotada, así pues

$$-P_{ij}^{max} \leq B_{ij}(\delta_i - \delta_j) \leq P_{ij}^{max}, \quad \forall j \in \Omega_i, \ \forall i \tag{1.37}$$

Los elementos principales de este problema son:

1. **Datos**

   n: el número de generadores
   P_i^{min}: la potencia mínima del generador i
   P_i^{max}: la potencia máxima del generador i
   B_{ij}: la susceptancia de la línea $i - j$
   P_{ij}^{max}: la capacidad de transporte máxima de la línea $i - j$
   C_i: el coste de producción de potencia del generador i
   Ω_i: el conjunto de nudos conectados al nudo i
   D_i: la demanda en el nudo i

2. **Variables**

   p_i: la potencia que debe producir el generador i
   δ_i: el ángulo del nudo i

3. **Restricciones.** Las restricciones en este problema son

$$\begin{array}{rcl} \delta_k & = & 0 \\ \sum\limits_{j\in\Omega_i} B_{ij}(\delta_i - \delta_j) + p_i & = & D_i \quad i = 1, 2, \ldots, n \\ -P_{ij}^{max} \ \leq \ B_{ij}(\delta_i - \delta_j) & \leq & P_{ij}^{max}; \ \forall j \in \Omega_i, \ i = 1, 2, \ldots, n \\ P_i^{min} \ \leq \ p_i & \leq & P_i^{max}; \ i = 1, 2, \ldots, n \end{array} \tag{1.38}$$

4. **Función a minimizar.** El objetivo del problema del despacho económico es minimizar el precio total de la producción de potencia, que puede concretarse como

$$Z = \sum_{i=1}^{n} C_i \, p_i \tag{1.39}$$

   donde C_i es el precio de la producción del generador i, y n el número de generadores.

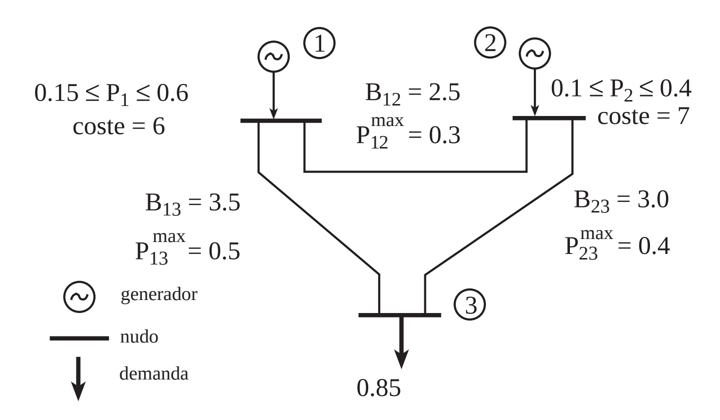


Figura 1.5: Ejemplo de despacho económico.

Ejemplo 1.7 (problema del despacho económico). Considérese un sistema de 3 nudos y 3 líneas (véase la figura 1.5). El generador del nudo 1 produce al precio de 6 y sus límites inferior y superior son, respectivamente, 0.15 y 0.6. El coste de producción del generador del nudo 2 es 7 y sus cotas superior e inferior son, respectivamente, 0.1 y 0.4. La línea 1–2 tiene susceptancia 2.5 y admite un transporte máximo de 0.3; la línea 1–3 tiene susceptancia 3.5 y cota superior de transporte de 0.5; finalmente la línea 2–3 tiene susceptancia 3.0 y límite superior de transporte 0.4. El sistema tiene un único nudo de demanda en el nudo 3 con un valor de 0.85. Se considera un periodo de una hora. El origen se sitúa en el nudo 3.

El problema del despacho económico para este ejemplo tiene la forma siguiente. Minimizar

$$6p_1 + 7p_2 \tag{1.40}$$

sujeto a

$$\begin{array}{rcl} \delta_3 & = & 0 \\ 3.5(\delta_3 - \delta_1) + 2.5(\delta_2 - \delta_1) + p_1 & = & 0 \\ 3.0(\delta_3 - \delta_2) + 2.5(\delta_1 - \delta_2) + p_2 & = & 0 \\ 3.5(\delta_1 - \delta_3) + 3.0(\delta_2 - \delta_3) & = & 0.85 \\ 0.15 \ \leq \ p_1 & \leq & 0.6 \\ 0.10 \ \leq \ p_2 & \leq & 0.4 \\ -0.3 \ \leq \ 2.5(\delta_1 - \delta_2) & \leq & 0.3 \\ -0.4 \ \leq \ 3.0(\delta_2 - \delta_3) & \leq & 0.4 \\ -0.5 \ \leq \ 3.5(\delta_1 - \delta_3) & \leq & 0.5 \end{array} \tag{1.41}$$

Las variables a optimizar son p_1, p_2, δ_1 y δ_2.

La solución óptima para este problema de despacho económico, que será resuelto en la sección 11.2.7, es:

$$Z = 5.385, \qquad \mathbf{p} = (0.565, 0.285)^T, \qquad \boldsymbol{\delta} = (-0.143, -0.117, 0)^T$$

El generador 1 debe producir 0.565 y el generador 2, 0.285. ■

Ejercicios

1.1 Pedro Pérez fabrica cable eléctrico de alta calidad usando dos tipos de aleaciones metálicas, A y B. La aleación A contiene un 80% de cobre y un 20% de aluminio, mientras que la B incluye un 68% de cobre y un 32% de aluminio. La aleación A tiene un precio de 80 euros por tonelada, y la B, 60 euros por tonelada. ¿Cuáles son las cantidades que Pedro Pérez debe usar de cada aleación para producir una tonelada de cable que contenga al menos un 20% de aluminio y cuyo coste de producción sea el menor posible?

1.2 Tres empleados deben realizar seis tareas distintas. El empleado i puede hacer a_{ij} partes de la tarea j en una hora y se le paga c_{ij} por hora. El número total de horas de trabajo para el empleado i es b_{1i} y el número de unidades que requiere la tarea j es b_{2j}. Se desea determinar el plan de trabajo que da lugar a un coste mínimo

$$C = \sum_{i=1}^{3} \sum_{j=1}^{6} c_{ij} x_{ij}$$

donde x_{ij} representa el número de horas empleadas en la tarea j por el empleado i. Plantéese este problema como un PPL.

1.3 Una compañía de fabricación de muebles ha de determinar cuántas mesas, sillas, pupitres y librerías debe hacer para optimizar el uso de sus recursos. Estos productos utilizan dos tipos diferentes de paneles, y la compañía dispone de 1500 tableros de un tipo y 1000 de otro tipo. Por otro lado cuenta con 800 horas de mano de obra. Las predicciones de venta así como los pedidos atrasados exigen la fabricación de al menos 40 mesas, 130 sillas, 30 pupitres y como máximo 10 librerías. Cada mesa, silla, pupitre y librería necesita $5, 1, 9,$ y 12 tableros, respectivamente, del primer tipo de panel y $2, 3, 4,$ y 1 tableros del segundo. Una mesa requiere 3 horas de trabajo; una silla, 2; un pupitre, 5; y una librería 10. La compañía obtiene un beneficio de 12 dólares en cada mesa, 5 dólares en cada silla, 15 dólares en un pupitre, y 10 dólares en una librería. Plantéese el modelo de programación lineal para maximizar los beneficios totales. Modifíquese el problema para imponer que deban fabricarse cuatro sillas por cada mesa.

1.4 Una empresa que produce un cierto producto P consta de dos plantas. Cada planta produce 90 toneladas de P al mes, y el producto se distribuye en tres mercados distintos. La tabla 1.3 muestra los precios unitarios del envío de una tonelada de P desde cada planta a cada mercado. La empresa desea enviar el mismo número de toneladas a cada mercado y minimizar el coste total. Formúlese el problema como un PPL.

1.5 Se está construyendo la carretera de la figura 1.6. Los trabajos en el terreno prosiguen entre dos puntos de la carretera. La figura 1.6 indica el

Tabla 1.3: El problema de la distribución

Plantas	Mercado_1	Mercado_2	Mercado_3
Planta_1	1	3	5
Planta_2	2	5	4

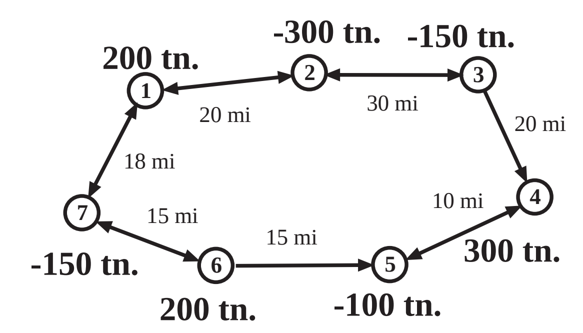


Figura 1.6: Diagrama de la carretera y los datos del transporte.

número de toneladas por nudo que debe transportarse. Si esta cantidad es positiva, el nudo es una fuente; en otro caso es un sumidero. El objetivo es minimizar el número total de toneladas que deben desalojarse. Plantéese el problema como un PPL.

1.6 Un productor de electricidad tiene que planificar la producción en cada hora para maximizar los beneficios vendiendo la energía en un horizonte temporal que abarca un número de horas dado. El productor no genera energía antes de dicho periodo. Los precios de la energía en cada hora pueden predecirse con garantías y se consideran conocidos. La energía mínima que el productor puede generar en cada hora es cero y el máximo es una cantidad fija. La diferencia de producción en horas consecutivas no puede exceder un determinado límite. El coste de generación de energía es lineal. Formúlese este problema como un PPL.

1.7 En el sistema de muelles de la figura 1.7, los puntos negros están fijos mientras los blancos están libres. Todos los muelles pueden rotar libremente alrededor de cada nudo. El comportamiento de cada muelle se caracteriza mediante una constante positiva k_i en las unidades apropiadas. La posición de equilibrio del nudo central se determina resolviendo el sistema

$$\sum_{i=1}^{4} k_i(x - x_i) = 0$$

si x_i es el vector que indica la posición del nudo i y x es la posición del nudo libre. Se desea encontrar la combinación de muelles que minimiza el

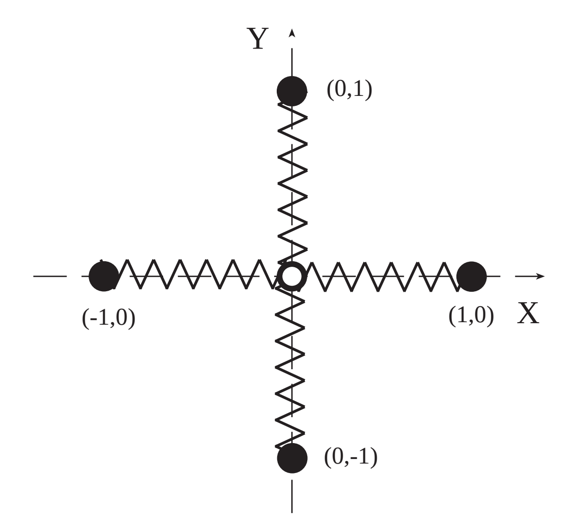


Figura 1.7: Sistema de muelles.

trabajo realizado por el nudo libre bajo la acción de una fuerza constante y conocida F, actuando en dicho nudo, considerando que las constantes de los cuatro muelles están restringidas por la condición lineal

$$\sum_i k_i = c$$

donde c es una constante fija positiva. Plantéese el problema como un PPL.

Capítulo 2

Programación lineal entera-mixta

2.1 Introducción

En el capítulo 1 se analizaron problemas de programación lineal en los que las variables tomaban valores reales. Sin embargo, en muchos casos realistas, algunas de las variables no son reales sino enteras, o incluso están más restringidas siendo binarias, es decir, que toman exclusivamente los valores 0 ó 1. Se verá en el Capítulo 7 que el empleo de variables enteras hace más complejo el problema de programación lineal, debido a la ausencia de continuidad.

En este capítulo se dan algunos ejemplos de problemas lineales entero-mixtos (PLEM), y en algunos de ellos, se emplean variables binarias.

2.2 El problema de la mochila

Una clase importante de problemas de programación entera son aquellos en los que las variables pueden tomar solamente dos valores. Esta situación se puede modelar empleando *variables 0–1*. Cada valor se asocia a una de las posibilidades de una elección binaria:

$$x = \begin{cases} 1 & \text{si la situación tiene lugar} \\ 0 & \text{en otro caso} \end{cases}$$

Un problema clásico en el que aparecen estas variables es el problema de la mochila. Considérese un excursionista que debe preparar su mochila. Considérese asimismo que hay una serie de objetos de utilidad para el excursionista, pero que el excursionista sólo puede llevar un número limitado de objetos. El problema consiste en elegir un subconjunto de objetos de tal forma que se maximice la utilidad que el excursionista obtiene, pero sin rebasar su capacidad de acarrear objetos.

Esto problema consta de los siguientes elementos:

1. **Datos**

   n: número de objetos

   a_j: peso de cada objeto j

   c_j: utilidad de cada objeto j

   b: la capacidad máxima de la mochila (del excursionista)

2. **Variables**

$$x_j = \begin{cases} 1 & \text{si el objeto } j \text{ se mete en la mochila} \\ 0 & \text{si no se mete} \end{cases} \tag{2.1}$$

3. **Restricciones.** La capacidad máxima de la mochila no ha de excederse:

$$\sum_{j=1}^{n} a_j x_j \leq b$$

4. **Función a maximizar.** El objetivo de este problema es maximizar la utilidad, que se puede expresar como

$$Z = \sum_{j=1}^{n} c_j x_j$$

Ejemplo 2.1 (el armador). Un armador tiene un carguero con capacidad de hasta 700 toneladas. El carguero transporta contenedores de diferentes pesos para una determinada ruta. En la ruta actual el carguero puede transportar algunos de los siguientes contenedores:

Contenedor	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8	c_9	c_{10}
Peso	100	155	50	112	70	80	60	118	110	55

El analista de la empresa del armador ha de determinar el envío (conjunto de contenedores) que maximiza la carga transportada.

Este problema puede formularse como un problema tipo mochila. Las variables son:

$$x_j = \begin{cases} 1 & \text{si el contenedor } j \text{ se carga} \\ 0 & \text{si no se carga} \end{cases}$$

La función objetivo es maximizar la carga que transportará el carguero:

$$\begin{aligned} Z \quad = \quad & 100x_1 + 155x_2 + 50x_3 + 112x_4 + 70x_5 \\ & +80x_6 + 60x_7 + 118x_8 + 110x_9 + 55x_{10} \end{aligned}$$

y la restricción es que el peso de la carga transportada no puede exceder la capacidad máxima del carguero:

$$\begin{aligned} 100x_1 + 155x_2 + 50x_3 + 112x_4 + 70x_5 + 80x_6 \\ +60x_7 + 118x_8 + 110x_9 + 55x_{10} \leq 700 \end{aligned}$$

Téngase en cuenta que $a_i = c_i; \forall i$, ya que la utilidad en este caso es el peso.

La decisión óptima es transportar los contenedores: $c_1, c_3, c_4, c_5, c_6, c_7, c_8, c_9$. El valor óptimo es 700, lo que indica que el carguero está completo.

■

2.3 Identificación de síntomas relevantes

Sea $\mathcal{D} = \{D_1, D_2, \ldots, D_n\}$ un conjunto conocido de posibles enfermedades. Considérese que los médicos, al identificar las enfermedades asociadas a un conjunto de pacientes, basan su decisión normalmente en un conjunto de síntomas $\mathcal{S} = \{S_1, S_2, \ldots, S_m\}$. Considérese que se quiere identificar un número mínimo de síntomas $\mathcal{S}_a \subset \mathcal{S}$, de tal manera que cada enfermedad puede distinguirse perfectamente de las otras de acuerdo con los niveles de los síntomas en el conjunto $\mathcal{S}_a$. Determinar el número mínimo de síntomas es importante ya que da lugar a un coste mínimo de diagnóstico.

Este problema consta de los siguientes elementos:

1. **Datos**

   $\mathcal{D}$: conjunto de enfermedades

   $\mathcal{S}$: conjunto de síntomas

   n: número de enfermedades (cardinal de $\mathcal{D}$)

   m: número de síntomas (cardinal de $\mathcal{S}$)

   c_{ij}: el nivel del síntoma j asociado a la enfermedad i

   d_{ikj}: nivel de discrepancia entre las enfermedades i y k debido al síntoma j

   a: nivel mínimo requerido de discrepancia (se explica en lo que sigue)

2. **Variables**

$$x_j = \begin{cases} 1 & \text{si el síntoma } j \text{ pertenece a } \mathcal{S}_a \\ 0 & \text{si no pertenece} \end{cases} \tag{2.2}$$

3. **Restricciones.** El subconjunto $\mathcal{S}_a$ debe ser suficiente para distinguir claramente todas las enfermedades:

$$\sum_{j=1}^{m} x_j d_{ikj} \geq a; \ \ \forall i, k \in \{1, 2, \ldots, n\}, \ i \neq k \tag{2.3}$$

donde

$$d_{ikj} = \begin{cases} 1 & \text{si} \quad c_{ij} \neq c_{kj} \\ 0 & \text{si} \quad c_{ij} = c_{kj} \end{cases} \tag{2.4}$$

mide la discrepancia entre las enfermedades D_i y D_k en términos de los síntomas incluidos en $\mathcal{S}_a$, y $a > 0$ es el nivel de discrepancia deseado. Téngase en cuenta que a medida que el valor de a es mayor, mayor será el número de síntomas requeridos (cardinal de $\mathcal{S}_a$). En este caso

$$\sum_{j=1}^{m} x_j d_{ikj}$$

coincide con el número de síntomas en $\mathcal{S}_0$ que toman distintos valores para las enfermedades D_i y D_k, y a es el número mínimo, para cualquier par (D_i, D_k) de enfermedades, necesario para tener un subconjunto aceptable $\mathcal{S}_a$. Esto quiere decir que pueden desconocerse $a - 1$ síntomas, y aún así se puede diferenciar cualquier par de enfermedades (D_i, D_k).

4. **Función a minimizar.** El objetivo de este problema es minimizar el número de síntomas seleccionados, el cardinal del conjunto $\mathcal{S}_0$:

$$Z = \sum_{j=1}^{m} x_j.$$

El problema así formulado, nos permite determinar el conjunto mínimo $\mathcal{S}_0$, asociado a $a = 0$, de síntomas del conjunto $\mathcal{S}$ que permite identificar las enfermedades del conjunto $\mathcal{D}$. Sin embargo, si las enfermedades han de identificarse con alguna carencia de información, el conjunto $\mathcal{S}_0$ puede resultar inservible. Por tanto, normalmente se emplea un valor $a > 0$.

Una vez seleccionados los síntomas relevantes para identificar cada enfermedad, pueden determinarse los síntomas relevantes asociados a la enfermedad i. Esto puede hacerse minimizando

$$Z = \sum_{j=1}^{m} x_j$$

sujeto a

$$\sum_{j=1}^{m} x_j d_{ikj} > a; \quad k \in \{1, 2, \ldots, n\}, \ i \neq k \tag{2.5}$$

En otras palabras, se determina el subconjunto mínimo del conjunto $\mathcal{S}_{ai} \subseteq \mathcal{S}$ de manera que la enfermedad i tenga síntomas diferentes comparada con el resto de enfermedades. Este subconjunto se denomina *conjunto de síntomas relevantes* para la enfermedad i.

Ejemplo 2.2 (identificación de síntomas relevantes). Considérese el conjunto de enfermedades $\mathcal{D} = \{D_1, D_2, D_3, D_4, D_5\}$ y el conjunto de síntomas

Tabla 2.1: Síntomas asociados a todas las enfermedades del ejemplo 2.2

	Síntoma							
Enfermedad	S_1	S_2	S_3	S_4	S_5	S_6	S_7	S_8
D_1	2	3	1	1	1	2	1	2
D_2	1	1	1	1	3	1	2	1
D_3	3	4	2	3	2	2	3	2
D_4	2	2	2	2	2	1	2	3
D_5	1	1	1	2	1	1	1	2

Tabla 2.2: Síntomas relevantes para todas las enfermedades en el ejemplo 2.2 para $a = 1$

Enfermedad	Síntomas relevantes
D_1	$\{2\}$
D_2	$\{5\}$
D_3	$\{2\}$
D_4	$\{2\}$
D_5	$\{2, 5\}$

$\mathcal{S} = \{S_1, S_2, \ldots, S_8\}$. Considérese asimismo que los síntomas asociados a las diferentes enfermedades son los que aparecen en la Tabla 2.1.

Por tanto, minimizando la suma $Z = \sum_{j=1}^{m} x_j$ sujeta a (2.3), y dos valores de a, se concluye que el conjunto de síntomas $\{2, 5\}$ es un subconjunto mínimo y suficiente de síntomas que permite distinguir las 5 enfermedades. Sin embargo, si se emplea un nivel de discrepancia de $a = 3$, el conjunto mínimo requerido es $\{1, 2, 4, 5, 7\}$. Téngase en cuenta que en este caso es posible hacer un diagnóstico correcto aún en la ausencia de dos síntomas.

Finalmente, la Tabla 2.2 muestra el conjunto requerido de síntomas relevantes para cada enfermedad y $a = 1$. Obsérvese en la Tabla 2.1 que el síntoma 2 es suficiente para identificar las enfermedades D_1, D_3, y D_4, y que el síntoma 5 es suficiente para identificar la enfermedad D_2. Sin embargo, son necesarios los síntomas 2 y 5 para identificar la enfermedad D_5.

■

2.4 El problema de la Academia de Ingeniería

La academia de Ingeniería tiene m miembros y ha de seleccionar r nuevos miembros de los pertenecientes al conjunto de candidatos J. Con este fin, cada miembro puede apoyar un número mínimo de 0 y un máximo de r candidatos. Los r candidatos con mayor número de apoyos se incorporan a la academia.

Previo al proceso final de selección, se lleva a cabo un ensayo para determinar

el grado a apoyo a cada candidato. En este proceso, cada candidato puede asignar las puntuaciones en la lista **p** a un máximo de S candidatos, pero puede no asignar todas las puntuaciones.

Únicamente se conoce la suma de las puntuaciones de cada candidato. El problema a resolver consiste en determinar el valor mínimo y máximo de los apoyos finales a cada candidato, basándose en los resultados de la prueba, considerando que asignar una puntuación a un candidato implica que el miembro que la asigna apoya al candidato.

Este problema consta de los siguientes elementos:

1. **Data**

   I: número de miembros de la Academia de Ingeniería

   J: número de candidatos

   S: número de puntuaciones distintas que pueden asignarse

   p_s: la puntuación s

   C_j: la puntuación total asociada al candidato j

2. **Variables**

   x_{ijs}: variable binaria que toma el valor 1 si el miembro i asigna una puntuación p_s al candidato j; en otro caso, toma el valor 0

3. **Restricciones**

   - Cada miembro asigna como mucho una puntuación a cada candidato:

   $$\sum_{s=1}^{S} x_{ijs} \leq 1; \ \forall i \in \{1, 2, \ldots, I\}, \ j \in \{1, 2, \ldots, J\}$$

   - Cada miembro puede asignar la puntuación p_s como mucho a un candidato:

   $$\sum_{j=1}^{J} x_{ijs} \leq 1; \ \forall i \in \{1, 2, \ldots, I\}, \ s \in \{1, 2, \ldots, S\}$$

   - La puntuación total de cada candidato es un valor dado:

   $$\sum_{i=1}^{I} \sum_{s=1}^{S} p_s x_{ijs} = C_j; \ \forall j \in \{1, 2, \ldots, J\}$$

4. **Función a optimizar.** El objetivo de este problema consiste en minimizar y maximizar para cada candidato la siguiente función:

$$Z_j = \sum_{i=1}^{I} \sum_{s=1}^{S} x_{ijs}, \quad j \in \{1, 2, \ldots, J\} \tag{2.6}$$

Tabla 2.3: Puntuaciones totales recibidas por los 8 candidatos en el ejemplo 2.3

Candidato	1	2	3	4	5	6	7	8
Puntuación recibida	71	14	139	13	137	18	24	8

Ejemplo 2.3 (problema de la Academia de Ingeniería). Considérese que la Academia de Ingeniería tiene 20 miembros y que $r = 4$ nuevos miembros han de ser seleccionados entre $J = 8$ candidatos, y que $\mathbf{p} \equiv \{10, 8, 3, 1\}$, lo que implica $S = 4$.

La información disponible es la que aparece en la última fila de la Tabla 2.3, esto es, la puntuación total que recibe cada candidato en la primera ronda; y nos interesa determinar el número de apoyos de cada candidato (obsérvese la penúltima fila de la Tabla 2.4).

Las puntuaciones reales recibidas por cada candidato por parte de cada miembro aparecen en la Tabla 2.4 (téngase en cuenta que esta información no está disponible, pero se muestra a efectos de ilustrar el problema).

Si se minimiza y maximiza (2.6) para todos los candidatos, se obtienen los resultados que aparecen en la Tabla 2.5, de la que se derivan las conclusiones siguientes:

1. Solamente los candidatos 3 y 5 tienen como mínimo 15 apoyos garantizados. Obsérvese que el siguiente, el candidato 1, cuenta solamente con 8 apoyos garantizados.

2. Observando la Tabla 2.5, no está claro que los candidatos 3 y 5 vayan a ser miembros de la Academia, dado que los candidatos $6, 1$ y 7 tienen un máximo de $18, 20$ y 20 apoyos garantizados, y pueden obtener solamente $15, 16$ ó 17 apoyos.

3. Para conocer, antes de la elección final, si el candidato 3 será miembro de la Academia, se necesita añadir restricciones al problema. Por ejemplo, añadiendo que los números totales de apoyos a los candidatos $1, 5, 6,$ y 7 han de ser mayores que el número total de apoyos al candidato 3 resulta:

$$\begin{array}{rcl}
\sum_{i=1}^{I}\sum_{s=1}^{S} x_{i1s} & \geq & \sum_{i=1}^{I}\sum_{s=1}^{S} x_{i3s} \\
\sum_{i=1}^{I}\sum_{s=1}^{S} x_{i5s} & \geq & \sum_{i=1}^{I}\sum_{s=1}^{S} x_{i3s} \\
\sum_{i=1}^{I}\sum_{s=1}^{S} x_{i6s} & \geq & \sum_{i=1}^{I}\sum_{s=1}^{S} x_{i3s} \\
\sum_{i=1}^{I}\sum_{s=1}^{S} x_{i7s} & \geq & \sum_{i=1}^{I}\sum_{s=1}^{S} x_{i3s}
\end{array}$$

Dado que esto da lugar a un problema infactible, se puede asegurar que el candidato 3 será miembro de la Academia de Ingeniería.

■

Tabla 2.4: Puntuaciones recibidas por los 8 candidatos en el ejemplo 2.3

	Candidato							
Miembro	1	2	3	4	5	6	7	8
1	3	–	10	–	8	1	–	–
2	1	–	10	–	8	3	–	–
3	–	1	–	3	10	–	8	–
4	–	3	10	–	8	1	–	–
5	3	–	8	–	10	–	1	–
6	1	–	10	–	8	–	3	–
7	10	–	8	–	3	1	–	–
8	3	–	10	1	8	–	–	–
9	8	–	3	–	10	1	–	–
10	–	3	10	–	1	–	8	–
11	8	–	1	–	10	–	3	–
12	–	–	–	–	10	–	–	–
13	–	–	10	–	8	–	–	–
14	10	–	–	1	3	–	–	8
15	3	–	10	–	8	–	1	–
16	10	–	1	–	8	–	3	–
17	1	3	10	8	–	–	–	–
18	1	3	8	–	10	–	–	–
19	1	–	10	–	3	8	–	–
20	8	1	10	–	3	–	–	–
Número de apoyos	15	6	17	4	19	7	6	1
Puntuación total	71	14	139	13	137	18	24	8

Tabla 2.5: Cota superior e inferior y número real de apoyos a los candidatos en el ejemplo 2.3

	Candidato							
Apoyos	1	2	3	4	5	6	7	8
Mínimo	8	3	15	2	15	2	3	1
Máximo	20	14	20	13	20	18	20	8
Real	15	6	17	4	19	7	6	1
Puntuaciones	71	14	139	13	137	18	24	8

2.5 El problema del horario

Este ejemplo constituye una formulación sencilla del "problema del horario escolar". Su objetivo es asociar aulas y horas a las asignaturas de un programa académico dividido en cursos.

Se considera que están disponibles n_c aulas y n_h horas, respectivamente, para enseñar n_s asignaturas. Estas asignaturas están agrupadas por (1) cursos y (2) profesores. La variable binaria $v(s,c,h)$ es igual a 1 si la asignatura s se enseña en la clase c a la hora h, y 0 en otro caso.

Se denomina Ω, al conjunto de todas las asignaturas, Ω_i, al conjunto de las n_i asignaturas enseñadas por el profesor i, y Δ_b, al conjunto de las n_b asignaturas agrupadas en el curso académico b. Los índices s, c, h, i, y b indican respectivamente asignatura, clase, hora, profesor y bloque.

Este problema consta de los siguientes elementos:

1. **Data**

   n_c: número de aulas

   n_h: número de horas

   n_s: número de asignaturas

   n_i: número de asignatura que ha de impartir el profesor i

   n_b: número de cursos

   Ω: conjunto de todas las asignaturas

   Ω_i: conjunto de asignaturas que ha de impartir el profesor i

   Δ_b: conjunto de asignaturas del curso b

2. **Variables**

   $v(s,c,h)$: variable binaria que toma el valor 1 si la asignatura s se imparte en el aula c a la hora h, y 0 en otro caso

3. **Restricciones**

   (a) Cada profesor imparte todas sus asignaturas:

$$\sum_{s \in \Omega_i} \sum_{c=1}^{n_c} \sum_{h=1}^{n_h} v(s,c,h) = n_i, \qquad \forall i \tag{2.7}$$

   (b) Cada profesor imparte como mucho una asignatura cada hora:

$$\sum_{s \in \Omega_i} \sum_{c=1}^{n_c} v(s,c,h) \leq 1, \qquad \forall h, \quad \forall i \tag{2.8}$$

   (c) Cada asignatura se imparte una sola vez:

$$\sum_{c=1}^{n_c} \sum_{h=1}^{n_h} v(s,c,h) = 1, \qquad \forall s \tag{2.9}$$

(d) En cada clase y hora se imparte como mucho una sola asignatura:

$$\sum_{s \in \Omega} v(s,c,h) \leq 1, \qquad \forall c, \quad \forall h \tag{2.10}$$

(e) En cada hora, se enseña como mucho una asignatura de cada curso:

$$\sum_{s \in \Delta_b} \sum_{c=1}^{n_c} v(s,c,h) \leq 1, \qquad \forall h, \quad \forall b \tag{2.11}$$

4. **Función a optimizar.** No es sencillo establecer qué función objetivo minimizar. En este ejemplo se emplea una sencilla. El objetivo es lograr un horario compacto. Esto se logra minimizando

$$\sum_{s \in \Omega} \sum_{c=1}^{n_c} \sum_{h=1}^{n_h} (c+h) \ v(s,c,h)$$

sujeto a las restricciones (2.7)–(2.10).

Se ha elegido esta función objetivo dado que penaliza el que las variables $v(s,c,h)$ tomen el valor 1 para valores elevados de c y h. Por tanto, su objetivo es compactar el horario. Se hace que los números que identifican la aulas y las horas sean lo más bajos posibles.

Ejemplo 2.4 (problema del horario). Considérense 3 aulas, 5 horas, 8 asignaturas, 2 profesores, y 2 cursos. El conjunto de todas las asignaturas es $\Omega = \{s_1, s_2, \ldots, s_8\}$, el conjunto de las asignaturas del profesor 1 es $\Omega_1 = \{s_1, s_2, s_8\}$, el conjunto de las asignaturas del profesor 2 es $\Omega_2 = \{s_3, s_4, s_5, s_6, s_7\}$, el conjunto de asignaturas del curso 1 es $\Delta_1 = \{s_1, s_2, s_3, s_4\}$, y el conjunto de las asignaturas del curso 2 es $\Delta_2 = \{s_5, s_6, s_7, s_8\}$. Téngase en cuenta que $\Omega_1 \cup \Omega_2 = \Omega$ y $\Omega_1 \cap \Omega_2 = \emptyset$, y $\Delta_1 \cup \Delta_2 = \Omega$ y $\Delta_1 \cap \Delta_2 = \emptyset$.

La solución se muestra en la tabla siguiente:

	$h=1$	$h=2$	$h=3$	$h=4$	$h=5$
$c=1$	s_7	s_6	s_3	s_4	s_5
$c=2$	s_2	s_1	s_8	–	–
$c=3$	–	–	–	–	–

El horario para el profesor 1 es

	$h=1$	$h=2$	$h=3$	$h=4$	$h=5$
$c=1$	–	–	–	–	–
$c=2$	s_2	s_1	s_8	–	–
$c=3$	–	–	–	–	–

El horario para el profesor 2 es

	$h=1$	$h=2$	$h=3$	$h=4$	$h=5$
$c=1$	s_7	s_6	s_3	s_4	s_5
$c=2$	–	–	–	–	–
$c=3$	–	–	–	–	–

El horario para el curso 1 es

	$h=1$	$h=2$	$h=3$	$h=4$	$h=5$
$c=1$	–	–	s_3	s_4	–
$c=2$	s_2	s_1	–	–	–
$c=3$	–	–	–	–	–

El horario para el curso 2 es

	$h=1$	$h=2$	$h=3$	$h=4$	$h=5$
$c=1$	s_7	s_6	–	–	s_5
$c=2$	–	–	s_8	–	–
$c=3$	–	–	–	–	–

■

2.6 Modelos de localización de plantas productivas

En este ejemplo se describe un modelo de *localización de plantas productivas*, o de forma más precisa, el *problema de localización de plantas productivas con capacidad limitada.* Se trata de elegir la localización de una serie de plantas de entre un conjunto dado de posibles localizaciones, teniendo en cuenta las necesidades de los consumidores y optimizando algún criterio económico. Normalmente, la construcción de una planta origina un coste importante que no depende del nivel de producción de esa planta.

Este problema tiene aplicación en campos diversos. Por ejemplo, pueden construirse centrales productivas en diversos lugares para maximizar el beneficio económico teniendo en cuenta los costes de producción y de transporte. La Figura 2.1 muestra una solución del problema de la localización de plantas para cubrir las necesidades de un conjunto de consumidores.

Por tanto, los elementos fundamentales de este problema son

1. **Datos**

   I: conjunto $\{1,\ldots,n\}$ de n consumidores

   J: conjunto $\{1,\ldots,m\}$ de m lugares donde las plantas pueden ser construidas

   f_j: coste fijo de construcción de la planta localizada en j para $j \in J$

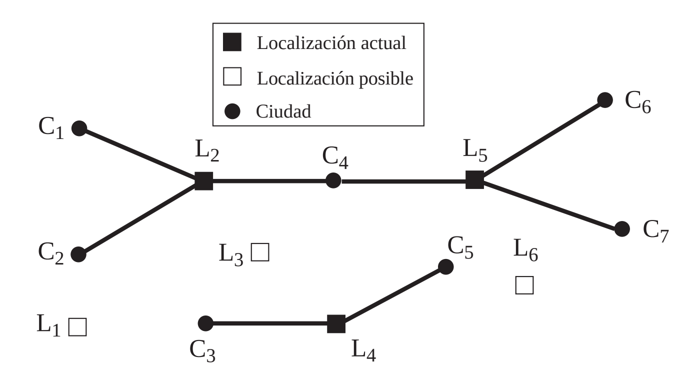


Figura 2.1: Solución del ejemplo de localización de plantas con capacidad limitada.

c_{ij}: beneficio unitario por venta, al consumidor i, de bienes producidos en la planta j. Normalmente, los costes c_{ij} dependen de los costes de producción en la planta j, la demanda y precio de venta al consumidor i, y el coste de transporte desde la planta j al consumidor i.

u_j: la capacidad productiva de la planta localizada en j

b_i: la demanda el consumidor i

2. **Variables.** Las variables de este problema son las siguientes:

y_j: variable binaria que permite modelar la "construcción" de una planta productiva en la localización j. Esto es:

$$y_j = \begin{cases} 1 & \text{si se construye la planta productiva } j \\ 0 & \text{en otro caso} \end{cases} \tag{2.12}$$

x_{ij}: cantidad de producto enviada desde la planta j al consumidor i.

3. **Restricciones.** Las restricciones de este problema son las siguientes. Ha de satisfacerse la demanda de cada consumidor:

$$\sum_{j \in J} x_{ij} = b_i, \quad \forall i \in I \tag{2.13}$$

Dado que al consumidor i no se le puede suministrar desde j a no ser que se haya construido una central en j, son necesarias las siguientes restricciones:

$$\sum_{i \in I} x_{ij} \leq u_j y_j, \quad \forall j \in J \tag{2.14}$$

Estas desigualdades lineales establecen que el consumidor i se puede suministrar desde j sólo si la planta se construye en j; dado que $y_j = 0$ implica

Tabla 2.6: Demandas de las ciudades

Ciudad	Demanda
C_1	1.5
C_2	2.0
C_3	3.0
C_4	4.0
C_5	2.5
C_6	1.0
C_7	2.0

que $x_{ij} = 0, \ \forall i$ y $y_j = 1$, da lugar a la restricción $\sum_{i \in I} x_{ij} \leq u_j$, que implica que la producción de la planta j no puede exceder su capacidad máxima. Además, las restricciones sobre las variables son

$$y_j \in \{0,1\}, \ \forall j \in J \tag{2.15}$$

$$x_{ij} \geq 0 \ \forall i \in I, \ \ \forall j \in J \tag{2.16}$$

4. **Función a optimizar.** En la denominada formulación estricta del problema de localización de plantas de tamaño cualquiera, se maximiza

$$Z = \sum_{i \in I} \sum_{j \in J} c_{ij} x_{ij} - \sum_{j \in J} f_j y_j \tag{2.17}$$

Con este modelo, se resuelve fácilmente el problema de localización de centrales de cualquier capacidad. De hecho, si se dispone de un conjunto factible de localizaciones, la solución consiste en asignar a cada consumidor la planta más rentable. Sin embargo, dado que no es realista suponer que a todos los consumidores se les pueda suministrar desde una única planta, es necesario considerar capacidades máximas.

Ejemplo 2.5 (localización de plantas industriales). Una empresa considera la construcción de plantas productivas para suministrar un determinado producto a 7 ciudades. La demanda de cada una de esas ciudades puede estimarse mediante factores demográficos y sociales. Estas estimaciones se muestran en la Tabla 2.6.

Un determinado estudio estadístico ha identificado 6 posibles localizaciones para las plantas industriales. Se supone que todas las plantas tienen las mismas características. La capacidad máxima de producción de cada planta es de 6 unidades. Se considera que el coste de recobrar la inversión a lo largo del horizonte de estudio de una planta es 10 unidades monetarias.

La Tabla 2.7 muestra el beneficio obtenido por vender a la ciudad i, una unidad de producto fabricado en la planta localizada en j.

Ha de determinarse el número de plantas y sus localizaciones, de forma tal que se suministre la demanda de las ciudades y el beneficio obtenido sea máximo.

Tabla 2.7: Beneficios en función de las localizaciones

	Ciudades (C_i)						
Localizaciones (L_j)	C_1	C_2	C_3	C_4	C_5	C_6	C_7
L_1	4.0	4.5	2.5	0.5	1.0	0.5	-3.5
L_2	4.0	4.5	2.5	4.2	3.5	1.5	-0.5
L_3	3.5	5.0	4.0	3.5	4.5	1.5	0.0
L_4	1.3	3.0	5.0	3.3	5.5	1.8	1.3
L_5	0.5	1.0	1.5	5.0	4.0	5.5	3.0
L_6	-1.0	0.0	1.5	3.3	4.0	4.5	2.0

El proceso de optimización asociado a este proceso de toma de decisión consiste en maximizar el beneficio total incluyendo costes de amortización, sujeto a las restricciones pertinentes. Por tanto, el problema consiste en maximizar

$$Z = \sum_{i=1}^{7} \sum_{j=1}^{6} c_{ij} x_{ij} - \sum_{j=1}^{6} 10 y_j$$

sujeto a

$$\begin{array}{rclrcl} \sum_{j=1}^{6} x_{1j} & = & 1.5; & \sum_{j=1}^{6} x_{2j} & = & 2.0 \\ \sum_{j=1}^{6} x_{3j} & = & 3.0; & \sum_{j=1}^{6} x_{4j} & = & 4.0 \\ \sum_{j=1}^{6} x_{5j} & = & 2.5; & \sum_{j=1}^{6} x_{6j} & = & 1.0 \\ \sum_{j=1}^{6} x_{7j} & = & 2.0 & & & \end{array} \tag{2.18}$$

y

$$\sum_{i=1}^{6} x_{ij} \leq 6 y_j; \quad j = 1, \ldots, 7 \tag{2.19}$$

$$\begin{array}{rcl} y_j & \in & \{0,1\}; \ j = 1, \ldots, 6 \\ x_{ij} & \geq & 0; \ i = 1, \ldots, 7; \ j = 1, \ldots, 6 \end{array} \tag{2.20}$$

donde (2.18) y (2.19) son las restricciones de demanda y de capacidad productiva, respectivamente.

La solución de este problema que se muestra en la Figura 2.1 consiste en emplazar 3 plantas industriales en las localizaciones L_2, L_4, y L_3; la distribución de producción por ciudades se muestra en la Tabla 2.8.

■

Tabla 2.8: Producción de cada planta que se suministra a cada ciudad

	Ciudades						
Localizaciones	C_1	C_2	C_3	C_4	C_5	C_6	C_7
L_2	1.5	2.0		1.0			
L_4			3.0		2.5		
L_5				3.0		1.0	2.0

2.7 Programación de centrales térmicas de producción de electricidad

El coste de poner en funcionamiento una central térmica de producción de energía eléctrica, habiendo estado parada un par de días, es del orden del coste de compra de un apartamento en una buena zona residencial de una ciudad media. Por tanto, la planificación de los arranques y paradas de las centrales térmicas ha de hacerse con cuidado. El problema de la programación horaria de centrales térmicas consiste en determinar para un horizonte de planificación multi-horario, el arranque y parada de cada central, de tal forma que se suministre la demanda en cada hora, el coste se minimice, y se satisfagan determinadas restricciones técnicas y de seguridad.

Un típico horizonte de planificación es un día dividido en horas. Si los intervalos horarios se denotan mediante k, el horizonte de planificación consta de los periodos

$$k = 1, 2, \ldots, K \tag{2.21}$$

donde K es típicamente igual a 24.

El coste de arranque de una central es una función exponencial del tiempo que la central lleva parada, pero se considerará constante (lo que es una simplificación razonable en la mayoría de los casos). Cada vez que una central se arranca se origina un gasto, lo que se puede expresar de la siguiente manera

$$C_j y_{jk} \tag{2.22}$$

donde C_j es el coste de arranque de la central j e y_{jk} es una variable binaria que toma el valor 1 si la central j se arranca al comienzo del periodo k y 0, en otro caso.

El coste de parada puede expresarse de forma análoga al coste de arranque, por tanto

$$E_j z_{jk} \tag{2.23}$$

donde E_j es el coste de la central j y z_{jk} una variable binaria que toma el valor 1 si la central j se para al comienzo del periodo k, y 0 en otro caso.

El coste de funcionamiento consta de un coste fijo y un coste variable. El coste fijo se puede expresar como

$$A_j\ v_{jk}, \tag{2.24}$$

donde A_j es el coste fijo de la central j y v_{jk} es una variable binaria que toma el valor 1 si la central j está arrancada durante el periodo k y 0, en otro caso.

El coste variable puede considerarse proporcional a la producción de la central:[1]

$$B_j\ p_{jk} \tag{2.25}$$

donde B_j es el coste variable de la central j y p_{jk} la producción de la central j durante el periodo k.

Las centrales térmicas no pueden funcionar ni por debajo de una producción mínima, ni por encima de una producción máxima. Estas restricciones se pueden formular de la siguiente manera

$$\underline{P}_j\ v_{jk} \leq p_{jk} \leq \overline{P}_j\ v_{jk} \tag{2.26}$$

donde $\underline{P}_j$ y $\overline{P}_j$ son respectivamente las producciones mínima y máxima de la central j.

El término de la izquierda de la restricción anterior establece que si la central j está funcionando durante el periodo k ($v_{jk} = 1$), su producción ha de estar por encima de su producción mínima. De forma análoga, el término de la derecha de esta restricción hace que si la central j está funcionando durante el periodo k ($v_{jk} = 1$), su producción ha de estar por debajo de su producción máxima. Si $v_{jk} = 0$, la restricción anterior hace que $p_{jk} = 0$.

Al pasar de un periodo de tiempo al siguiente, cualquier central térmica no puede incrementar su producción por encima de un máximo, denominado rampa máxima de subida de carga. Esta restricción se expresa de la siguiente manera

$$p_{jk+1} - p_{jk} \leq S_j \tag{2.27}$$

donde S_j es la rampa máxima de subida de carga de la central j.

Para el primer periodo del horizonte de planificación las restricciones previas tienen la forma siguiente

$$p_{j1} - P_j^0 \leq S_j \tag{2.28}$$

donde P_j^0 es la producción de la central j en el periodo previo al de comienzo del horizonte de planificación.

Análogamente, ninguna central puede bajar su producción por encima de un máximo, que se denomina rampa máxima de bajada de carga. Por tanto

$$p_{jk} - p_{jk+1} \leq T_j \tag{2.29}$$

donde T_j es la rampa máxima de bajada de la central j.

[1] Un modelo más preciso requiere el empleo de funciones cuadráticas o cúbicas.

Para el primer periodo del horizonte de planificación la restricción anterior toma la forma

$$P_j^0 - p_{j1} \leq T_j \tag{2.30}$$

Cualquier central que está funcionando puede pararse pero no arrancarse, y análogamente cualquier central parada puede arrancarse pero no pararse. Esto se expresa de la manera siguiente

$$y_{jk} - z_{jk} = v_{jk} - v_{jk-1} \tag{2.31}$$

Para el periodo primero la restricción anterior se convierte en

$$y_{j1} - z_{j1} = v_{j1} - V_j^0 \tag{2.32}$$

donde V_j^0 es una variable binaria que toma el valor 1 si la central j está en funcionamiento en el periodo anterior al primero del horizonte de planificación, y 0 en otro caso. Se anima al lector a verificar estas condiciones mediante ejemplos.

La demanda debe suministrarse en cada periodo, por tanto

$$\sum_{j=1}^{J} p_{jk} = D_k \tag{2.33}$$

donde J es el número de centrales y D_k la demanda en el periodo k.

Por razones de seguridad, la potencia total disponible en centrales en funcionamiento debe ser mayor que la demanda en una determinada cantidad de reserva. Esto se formula de la siguiente manera.

$$\sum_{j=1}^{J} \overline{P}_j v_{jk} \geq D_k + R_k \tag{2.34}$$

donde R_k es la cantidad requerida de reserva (por encima de la demanda) en el periodo k.

Los componentes principales de este problema son:

1. **Datos**

   K: número de periodos de tiempo que tiene el horizonte temporal

   C_j: coste de arranque de la central j

   E_j: coste de parada de la central j

   A_j: coste fijo de la central j

   B_j: coste variable de la central j

   $\underline{P}_j$: producción mínima de la central j

   $\overline{P}_j$: producción máxima de la central j

S_j: rampa máxima de subida de carga de la central j

P_j^0: producción de la central j en el periodo anterior al del comienzo del horizonte de planificación

T_j: rampa máxima de bajada de carga de la central j

V_j^0: constante binaria que toma el valor 1 si la central j está funcionando en el periodo previo al de comienzo del horizonte de planificación, y 0, en otro caso

J: número de centrales de producción

D_k: demanda en el periodo k

R_k: reserva requerida en el periodo k

2. **Variables.** La variables de este problema son las siguientes:

   y_{jk}: variable binaria que toma el valor 1, si la central j se arranca al comienzo del periodo k y 0, en otro caso

   z_{jk}: variable binaria que toma el valor 1, si la central j se para al comienzo del periodo k, y 0, en otro caso

   v_{jk}: variable binaria que toma el valor 1, si la central j está en funcionamiento durante el periodo k y 0, en otro caso

   p_{jk}: producción de la central j durante el periodo k

3. **Restricciones.** Las restricciones de este problema son las siguientes. Cualquier central debe funcionar por encima de su producción mínima y por debajo de su producción máxima, por tanto

$$\underline{P}_j v_{jk} \leq p_{jk} \leq \overline{P}_j v_{jk} \quad \forall j, k \tag{2.35}$$

La restricciones de rampa de subida han de satisfacerse:

$$p_{jk+1} - p_{jk} \leq S_j, \ \ \forall j, k = 0, \ldots, K-1 \tag{2.36}$$

donde

$$p_{j0} = P_j^0$$

La restricciones de rampa de bajada han de satisfacerse:

$$p_{jk} - p_{jk+1} \leq T_j, \ \ \forall j, k = 0, \ldots, K-1 \tag{2.37}$$

La lógica de cambio de estado (de arranque a parada y viceversa) ha de preservarse, por tanto

$$y_{jk} - z_{jk} = v_{jk} - v_{jk-1}, \ \ \forall j, k = 1, \ldots, K \tag{2.38}$$

donde

$$v_{j0} = V_j^0, \ \ \forall j$$

La demanda ha de satisfacerse en cada periodo, por tanto

$$\sum_{j=1}^{J} p_{jk} = D_k, \quad \forall k \tag{2.39}$$

Finalmente y por razones de seguridad, la reserva ha de mantenerse en todos los periodos, por tanto

$$\sum_{j=1}^{J} \overline{P}_j \; v_{jk} \geq D_k + R_k, \quad \forall k. \tag{2.40}$$

4. **Función a minimizar.** El objetivo de la programación horaria de centrales de producción de energía eléctrica es minimizar los costes totales; esto objetivo es por tanto minimizar

$$Z = \sum_{k=1}^{K} \sum_{j=1}^{J} [A_j \; v_{jk} + B_j \; p_{jk} + C_j \; y_{jk} + E_j \; z_{jk}] \tag{2.41}$$

El problema formulado en (2.35)–(2.41) es una versión simplificada del problema de la programación horaria de centrales térmicas. Obsérvese que es un problema de programación lineal entera-mixta.

Ejemplo 2.6 (programación horaria de centrales). Se considera un horizonte de planificación de 3 horas. Las demandas en esas horas son respectivamente 150, 500, y 400. La reservas son respectivamente 15, 50, y 40. Se considera 3 centrales de producción de energía eléctrica. Los datos de esas centrales se muestran a continuación:

Tipo de central	1	2	3
Producción máxima	350	200	140
Producción mínima	50	80	40
Límite de rampa de subida	200	100	100
Límite de rampa de bajada	300	150	100
Coste fijo	5	7	6
Coste de arranque	20	18	5
Coste de parada	0.5	0.3	1.0
Coste variable	0.100	0.125	0.150

Se considera que todas las centrales están paradas en el periodo previo al primero del horizonte de planificación.

La producción óptima de cada central se muestra a continuación

	Hora		
Central	1	2	3
1	150	350	320
2	—	100	080
3	—	050	—
Total	150	500	400

El coste mínimo de producción es 191. La central 1 se arranca al comienzo de la hora 1 y permanece acoplada durante 3 horas. La central 2 se arranca al comienzo de la hora 2 y permanece acoplada durante las horas 2 y 3. La central 3 se arranca al comienzo de la hora 2 y se para al comienzo de la hora 3.

■

Ejercicios

2.1 Pepe construye dos tipos de transformadores eléctricos y tiene disponible 6 toneladas de material ferromagnético y 28 horas de tiempo de manufacturación. El transformador tipo 1 requiere 2 toneladas de material y 7 horas de trabajo, mientras que el transformador tipo 2 necesita 1 unidad de material y 8 horas de trabajo. Los precios de venta de los transformadores 1 y 2 son respectivamente 120 y 80 en miles de euros. ¿Cuántos transformadores de cada tipo ha de construir Pepe para maximizar sus beneficios? Resuélvase el problema analítica y gráficamente.

2.2 Considérese una red de carreteras que interconecta un conjunto de ciudades. Se trata de determinar el camino más corto entre dos ciudades. Se supone conocida la distancia entre cada dos ciudades directamente conectadas. Formular este problema como un problema de programación lineal entera-mixta (*Problema del camino más corto*).

2.3 Considérese un viajante que quiere determinar la ruta de coste mínimo que le permite visitar todas las ciudades de una determinada área geográfica y volver a la de origen. Formúlese este problema como un problema de programación lineal entera-mixta (*Problema del viajante*).

2.4 Considérese el problema de determinar el número máximo de caminos disjuntos (sin tramos comunes) en una red de comunicaciones entre el origen p y el destino q. Formúlese este problema como un problema de programación lineal entera-mixta.

2.5 Sea un gráfico conexo $\mathcal{G} = (\mathcal{N}, \mathcal{A})$, en el que $\mathcal{N}$ es el conjunto de nudos, $\mathcal{A}$ el conjunto de ramas, y $\mathcal{T} = \{t_1, t_2, \dots t_k\} \subset \mathcal{N}$, un conjunto especial "vértices extremos". El problema de la fiabilidad de esta red consiste en evaluar la probabilidad de que sea posible la comunicación entre los elementos de $\mathcal{T}$ bajo la hipótesis de fallos aleatorios en la red. Sea $p(e)$ la probabilidad de que el enlace e esté *operativo*. La fiabilidad de la red es la probabilidad de que todos los posibles pares de terminales estén conectados al menos por un camino operativo. Considérese que ha de diseñarse una red de coste mínimo y que satisfaga un nivel especificado de fiabilidad. Para este diseño, ha de elegirse un conjunto de caminos de entre los de la red original. Considérese en particular la red original de la Figura 2.2 con terminales $\mathcal{T} = \{\infty, \in\}$ y considérese que el nivel de fiabilidad deseado es 0.90. El coste y la fiabilidad de cada camino se muestran en la tabla

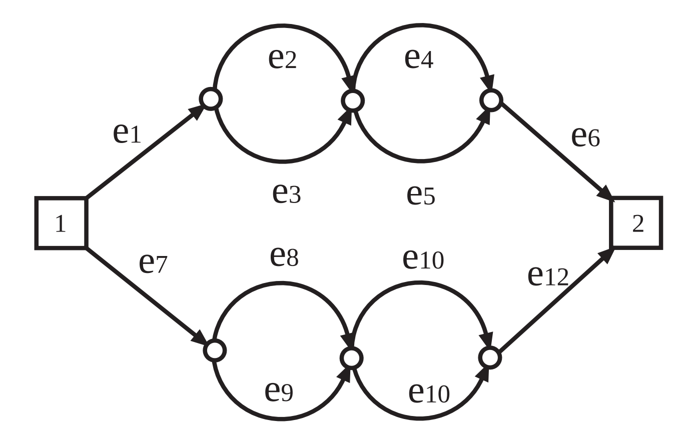


Figura 2.2: Topología de la red.

Tabla 2.9: Fiabilidad y coste de los enlaces de la red de la Figura 2.2

Enlace	Fiabilidad ($p(e_i)$)	Coste
e_1	0.90	1.90
e_2	0.85	1.85
e_3	0.95	1.95
e_4	0.70	1.70
e_5	0.80	1.80
e_6	0.90	1.90
e_7	0.90	1.90
e_8	0.50	1.35
e_9	0.60	1.45
e_{10}	0.60	1.20
e_{11}	0.30	1.30
e_{12}	0.90	1.90

2.9. Formúlese este problema como un problema de programación lineal entera-mixta.

2.6 Un productor de electricidad ha de planificar su producción horaria de energía para maximizar sus beneficios por venta de la misma en cada hora de su horizonte de planificación. A este respecto, formúlese un problema de programación lineal entera-mixta teniendo en cuenta lo siguiente.

(a) No hay producción antes del horizonte de planificación.

(b) Los precios horarios de venta se pueden predecir y por tanto se consideran conocidos.

(c) Si el productor produce, ha de hacerlo entre una producción máxima y una mínima que son conocidas. La producción mínima es mayor

Tabla 2.10: Tiempos que requieren los diferentes procesos

	Proceso				
Parte	L	S	D	M	G
A	0.6	0.4	0.1	0.5	0.2
B	0.9	0.1	0.2	0.3	0.3
Disponibilidad	10	3	4	6	5

que cero.

(d) La variación de la producción entre dos horas consecutivas no puede exceder un valor máximo conocido.

(e) El coste de producción es lineal.

2.7 Un productor de dos componentes, A y B, de una determinada máquina requiere llevar a cabo procesos L, S, D, M, y G. El tiempo requerido de proceso para cada parte, y los números totales de horas de proceso disponibles se muestran en la Tabla 2.10 (horas por unidad). Cada proceso puede ser utilizado 8 horas al día, y 30 días por mes.

(a) Determínese la estrategia óptima de producción que permite maximizar el número total de partes A y B en un mes.

(b) Si las cantidades de partes A y B han de ser iguales, ¿cuál es la estrategia óptima?

2.8 El gestor de un hospital debe planificar el horario de los trabajadores del mismo. Determínese el coste mínimo de personal para el hospital sabiendo que

(a) La jornada laboral consta de 3 turnos.

(b) En cada turno ha de haber al menos 1 médico, 3 enfermeras y 3 auxiliares de clínica.

(c) El número máximo de empleados que se requiere en cada turno es 10.

(d) Los salarios son los siguientes: 50 dólares/turno para un médico, 20 dólares/turno para un enfermero, y 10 dólares/turno para un auxiliar de clínica.

(e) El número total de empleados es: 15 médicos, 36 enfermeras, y 49 auxiliares de clínica.

(f) Cada empleado debe descansar al menos dos turnos consecutivos.

Capítulo 3

Programación no lineal

3.1 Introducción

Los capítulos 1 y 2 se dedicaron a los problemas de programación lineal, en los que las restricciones y la función a optimizar eran lineales.

Aunque los problemas de programación lineal son muy comunes y cubren un amplio rango de aplicaciones, en la vida real uno se tiene que enfrentar con cierta frecuencia a otro tipo de problemas que no son lineales. Cuando el conjunto de restricciones, la función objetivo, o ambos, son no lineales, se dice que se trata de un *problema de programación no lineal* (PPNL).

En este capítulo se presentan algunos problemas de programación no lineal. En algunos casos, coinciden con los que se han descrito en capítulos precedentes, pero bajo hipótesis distintas.

3.2 Algunos ejemplos geométricos

En esta sección, se describen algunos problemas de programación no lineal de carácter geométrico que pueden resolverse analíticamente.

3.2.1 El paquete postal

Un paquete postal es una caja de dimensiones x, y, y z (ver la Figura 3.1), que debe verificar los siguientes requisitos para ser aceptado en la oficina de correos. La altura más el perímetro de la base no puede exceder 108 cm

$$z + 2x + 2y \leq 108; \quad x, y, z \geq 0$$

pues no son posibles las longitudes negativas.

Se buscan las tres dimensiones que maximizan el volumen

$$V(x, y, z) = xyz$$

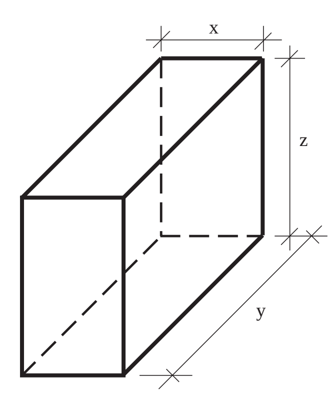


Figura 3.1: Esquema de las dimensiones del paquete postal.

3.2.2 La tienda de campaña

Una tienda tiene una base cuadrada de lado $2a$, cuatro paredes verticales de altura b, y un tejado que es un pirámide de altura h (ver la Figura 3.2). Si el volumen total V y la altura total H de la tienda están dados, el problema que se plantea es el de encontrar los valores óptimos a, b y h tales que la tienda resultante tenga una superficie mínima (se necesita un mínimo de material).

En este caso se pretende minimizar la superficie de la tienda. Como la superficie total es la suma de la superficie de las cuatro paredes más el tejado, se debe minimizar

$$S(a,b,h) = 4(2ab + a\sqrt{h^2 + a^2})$$

Las condiciones que deben exigirse se refieren al volumen total y la altura total; así, debe tenerse

$$\begin{array}{rcl} V & = & 4a^2\left(b + \frac{h}{3}\right) \\ H & = & b + h \end{array}$$

Además, todas la variables deben ser no negativas, es decir, $a \geq 0$, $b \geq 0$, $h \geq 0$.

3.2.3 La bombilla

Una bombilla está situada justo encima del centro de un círculo de radio r. Suponiendo que la intensidad de luz en cualquier punto de la circunferencia es proporcional a la raíz cuadrada del seno del ángulo con el que el rayo de luz llega a tal punto, y al inverso de la distancia a la bombilla, d, encontrar la altura óptima de la bombilla, de suerte que la intensidad en la circunferencia (frontera del círculo) sea máxima.

Se debe maximizar la intensidad de luz en los puntos de la circunferencia de radio $r = 30$ cm. Sea I la intensidad, medida en unidades apropiadas, que depende del seno del ángulo formado por el rayo de luz y el horizonte. En el

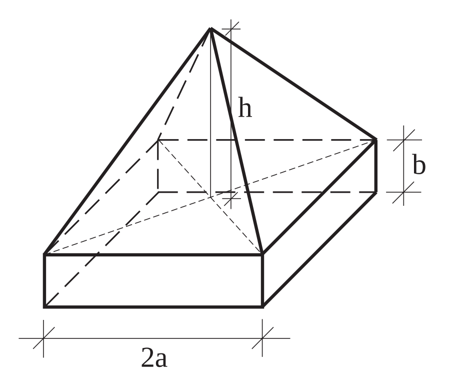


Figura 3.2: Esquema de la tienda de campaña.

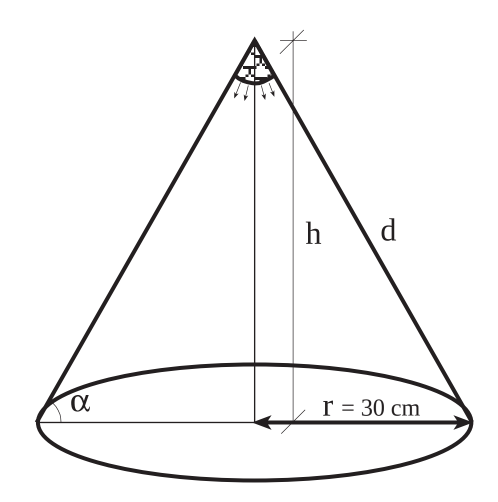


Figura 3.3: Esquema del ejemplo de la bombilla.

triángulo de la Figura 3.3 se ve que

$$\sin\alpha = \frac{h}{\sqrt{h^2 + r^2}}; \quad d = \sqrt{h^2 + r^2}$$

Así se puede escribir

$$I(h) = k\frac{h^{1/2}}{(h^2 + r^2)^{3/4}}$$

donde $k > 0$ es una constante de proporcionalidad. Esta expresión debe maximizarse con respecto a h, recordando que debe tenerse $h \geq 0$.

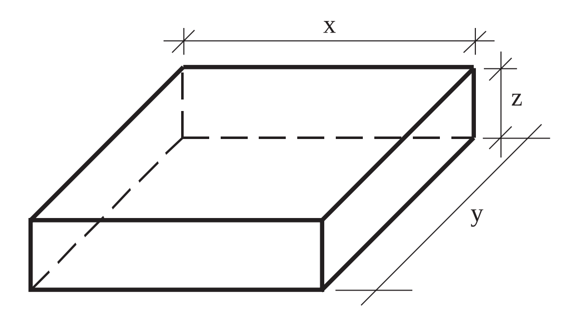


Figura 3.4: Boceto del problema del transporte de arena.

3.2.4 La superficie

Para encontrar el punto de la superficie tridimensional, $xyz = 1$, más cercano al origen, se debe minimizar la función (distancia al origen):

$$D(x, y, z) = \sqrt{x^2 + y^2 + z^2}$$

sujeto a la restricción de que el punto debe estar en la superficie:

$$xyz = 1$$

3.2.5 El transporte de arena

El coste de transportar arena de un lugar a otro en un contenedor de dimensiones x, y y z es 2 dólares por cada viaje completo (véase la Figura 3.4). Suponiendo que el precio del material de las paredes superior e inferior, y de los laterales del contenedor son el triple y el doble, respectivamente, de las paredes anterior y posterior, encontrar el precio mínimo de transportar $c = 50m^3$ de arena.

Si z es la altura del contenedor, el precio del mismo será

$$k(3xy + 2(2xz + 2yz) + xy)$$

donde $k = 4$ es una constante de proporcionalidad. Se debe añadir el precio del transporte en sí, que viene dado por

$$2\frac{50}{xyz}$$

El precio total, que debe minimizarse, es entonces

$$C(x, y, z) = k(3xy + 2(2xz + 2yz) + xy) + 2\frac{50}{xyz}$$

bajo las restricciones $x, y, z \geq 0$.

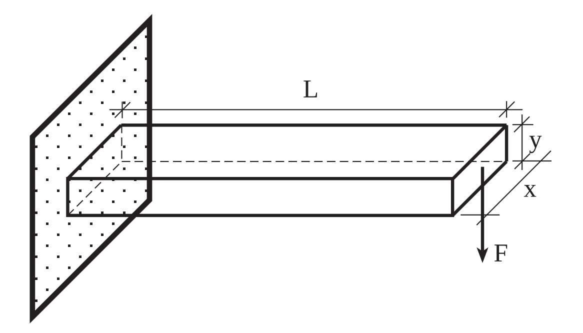


Figura 3.5: Voladizo.

3.3 Algunos ejemplos mecánicos

Otra categoría importante de ejemplos se refieren a situaciones que surgen en ingeniería mecánica. Se incluyen varias situaciones típicas que pueden darse en condiciones diversas. Todos ellos serán resueltos posteriormente.

3.3.1 El voladizo

Desea diseñarse un voladizo de sección rectangular y longitud dada para conseguir peso mínimo, y a la vez asegurar una deformación máxima transversal bajo la acción de una carga vertical actuando en el extremo libre. El material para construir el voladizo tiene una peso específico conocido.

Sean x e y la anchura y altura (véase la Figura 3.5) que se buscan. Sean L, F, S, y γ, la longitud, la carga en el extremo libre, la deformación máxima permitida, y el peso específico, respectivamente, esto es, el conjunto de datos de nuestro problema. El objetivo es minimizar el peso

$$W(x, y) = \gamma L x y.$$

Como la anchura x debe ser mayor que 0.5 y, de acuerdo con la teoría de *resistencia de materiales*, la deformación en el extremo libre viene dada por $FL^3/3EI$, donde E es el módulo de Young del material del voladizo, e $I = xy^3/12$ es el correspondiente momento de inercia de la sección rectangular, se tiene que minimizar $W(x, y)$ bajo las restricciones

$$\begin{array}{rcl} \dfrac{4FL^3}{Exy^3} & \leq & S \\ x & \geq & 0.5 \\ x, y & \geq & 0 \end{array}$$

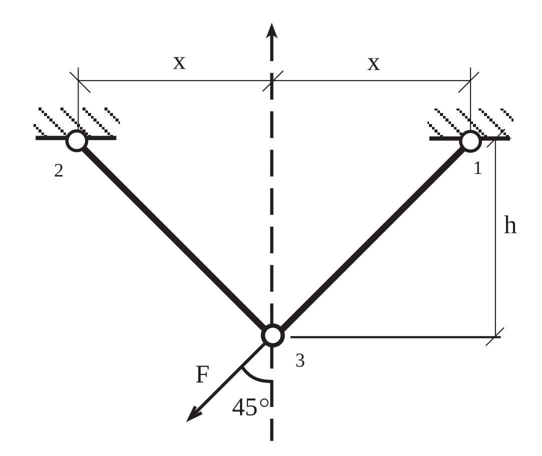


Figura 3.6: Estructura de dos barras usada en la sección 3.3.2.

3.3.2 La estructura de dos barras

Se desea diseñar la estructura con dos barras de la Figura 3.6 según tres objetivos distintos: peso mínimo, la tensión máxima no debe exceder un cierto umbral, y el desplazamiento en el pivote 3 no debe superar un cierto valor. El conjunto de datos del problema es:

γ: el peso específico del material de las barras

E: el módulo de Young del material de la estructura

F: la carga que actúa en el pivote fijo con un ángulo de 45° a la izquierda del eje Y

S_0: la tensión máxima admisible

D_0: el desplazamiento máximo admisible del pivote 3

h: la altura de la estructura

Las variables que se necesitan para determinar el diseño óptimo son

x: la distancia desde los pivotes fijos al eje Y

z: el área de la sección de los brazos de la estructura

D: el desplazamiento del pivote 3

S^1: la tensión en el pivote 1

S^2: la tensión en el pivote 2

W: el peso total de la de la estructura

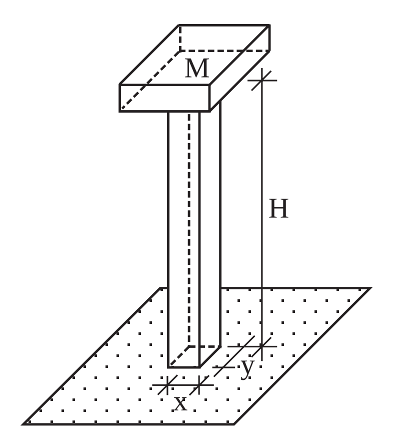


Figura 3.7: Columna sometida a pandeo.

Por tanto debe minimizarse

$$W(x,z) = 2\gamma z\sqrt{x^2+h^2} \tag{3.1}$$

bajo

$$
\begin{array}{rcll}
D(x,z) & = & \dfrac{F}{Eh^2 2\sqrt{2}} \dfrac{(h^2+x^2)^{3/2}(h^4+x^4)^{1/2}}{x^2 z} & \leq \; D_0 \\
S^1(x,z) & = & \dfrac{F}{2\sqrt{2}h} \dfrac{(x+h)\sqrt{x^2+h^2}}{xz} & \leq \; S_0 \\
S^2(x,z) & = & \dfrac{F}{2\sqrt{2}h} \dfrac{(h-x)\sqrt{x^2+h^2}}{xz} & \leq \; S_0 \\
x, z & \geq & 0 &
\end{array}
$$

3.3.3 La columna sometida a pandeo

Se desea diseñar una columna uniforme de sección rectangular y de altura dada que soporte una masa dada en su extremo. Por un lado, nos gustaría minimizar la cantidad de material que debe usarse, pero por otro, sería bueno maximizar la frecuencia natural de las vibraciones transversales. Se trata de encontrar las dimensiones óptimas de tal columna evitando el colapso debido a la compresión y al pandeo (fallo de estabilidad) (véase la Figura 3.7).

El conjunto de datos del problema es

M: la masa que debe soportar la columna

H: la altura de la columna

D: peso específico del material que debe usarse

E: el módulo de Young del mismo material

S: la máxima tensión permitida (fuerza por unidad de superficie)

Las variables de diseño son las dos dimensiones, x e y, de la sección transversal de la columna.

El primer objetivo consiste en minimizar la masa total de la columna

$$W(x, y) = DHxy$$

El segundo objetivo es maximizar la frecuencia de la vibración transversal que, según nos enseña la mecánica, viene dada por

$$\left(\frac{Exy^3}{4H^3(M + \frac{33}{140}DHxy)} \right)^{1/2}$$

Obsérvese que maximizar esta cantidad es equivalente a minimizar su recíproca.

Las restricciones que deben respetarse se refieren a las tensiones. Por un lado, la tensión no debe superar un cierto máximo S. Por otro lado, debe ser inferior a la tensión de pandeo. La tensión de compresión es Mg/xy donde g es la gravedad, mientras que la tensión de pandeo es $\pi^2 Ey^2/48H^2$. En definitiva deben exigirse las siguientes condiciones

$$\begin{array}{rcl} \dfrac{Mg}{xy} & \leq & S \\ \dfrac{Mg}{xy} & \leq & \dfrac{\pi^2 Ey^2}{48H^2} \\ x, y & \geq & 0 \end{array}$$

Suponiendo que ningún objetivo prevalece, debe minimizarse

$$Z = DHxy - \left(\frac{Exy^3}{4H^3 \left(M + \dfrac{33}{140} DHxy \right)} \right)^{1/2}$$

bajo las restricciones anteriores.

3.3.4 El sistema de vigas y cuerdas

En este ejemplo se considera una versión modificada del modelo de la Sección 1.7. Básicamente, se permite ahora que los datos dl_i, la distancia del punto de la viga b donde se aplica la carga i ($i \in \Omega_b$), sea una variable en vez de un dato. Para enfatizar este hecho se cambia la notación y se usa xl_i en vez de dl_i. Además se necesita un nuevo dato, l_b, la longitud total de la viga $b \in B$. Las

restricciones en esta nueva situación serían

$$\begin{array}{rcl}\sum\limits_{s\in\Psi_b} t_s &=& \sum\limits_{i\in\Omega_b} x_i + \sum\limits_{x\in\Theta_b} t_s, b\in B\\ \sum\limits_{s\in\Psi_b} dr_s t_s &=& \sum\limits_{i\in\Omega_b} xl_i x_i + \sum\limits_{x\in\Theta_b} dr_s t_s, b\in B\\ 0\le t_s &\le& T_s, s\in S\\ 0\le xl_i &\le& l_b, i\in\Omega_b\\ 0 &\le& x_i\end{array} \tag{3.2}$$

y la función a maximizar es

$$\sum_i x_i$$

En particular, si en el ejemplo concreto de la Figura 1.4 se permite que las cargas x_1 y x_2 se apliquen en los puntos que distan x_3 y x_4 desde el extremo izquierdo de las vigas 1 y 3, respectivamente, las ecuaciones de equilibrio se transforman en

$$\begin{array}{rcl}t_E+t_F &=& x_2\\ 10t_F &=& x_4x_2\\ t_C+t_D &=& t_F\\ 8t_D &=& 6t_F\\ t_A+t_B &=& x_1+t_C+t_D\\ 10t_B &=& 2t_C+10t_D+x_1x_3\end{array} \tag{3.3}$$

Si se expresan las tensiones en las cuerdas en términos de las variables independientes del problema, se concluye que se deben respetar las restricciones

$$\begin{array}{rcccl}t_F &=& \dfrac{x_2x_4}{10} &\le& 100\\ t_E &=& x_2-\dfrac{x_2x_4}{10} &\le& 100\\ t_D &=& \dfrac{3x_2x_4}{40} &\le& 200\\ t_C &=& \dfrac{x_2x_4}{40} &\le& 200\\ t_B &=& \dfrac{x_1x_3}{10}+\dfrac{2x_2x_4}{25} &\le& 300\\ t_A &=& x_1-\dfrac{x_1x_3}{10}+\dfrac{x_2x_4}{50} &\le& 300\\ 0 &\le& x_3 &\le& 10\\ 0 &\le& x_4 &\le& 10\\ 0 &\le& x_1 && \\ 0 &\le& x_2 &&\end{array} \tag{3.4}$$

que son no lineales. Así, se tiene un problema de programación no lineal.

La solución que se obtiene con GAMS es (véase la Sección 11.4.9)

$$Z = 700 \text{ en el punto } x_1 = 500;\ \ x_2 = 200;\ \ x_3 = 4.4;\ \ x_4 = 5.0$$

Las correspondientes tensiones en las cuerdas son

$$t_A = 300; \ \ t_B = 300; \ \ t_C = 25; \ \ t_D = 75; \ \ t_E = 100; \ \ t_F = 100$$

3.4 Algunos ejemplos de ingeniería eléctrica

Esta sección contiene algunos ejemplos con motivación eléctrica.

3.4.1 Estimación de estado en sistemas eléctricos

Los sistemas de energía eléctrica se extienden sobre naciones e incluso continentes para asegurar el suministro eléctrico a la industria, los negocios y los hogares. A lo largo de la red se encuentran situados aparatos de medida de diversos tipos.

Los voltímetros miden las tensiones con una cierta precisión. Se realiza una medida de tensión, que se designa por $\hat{v}_i$, en cada nudo i; su nivel de calidad viene determinado por el parámetro σ_i^v. Cuanto menor es σ_i^v, mayor es la precisión de la medida.

Normalmente, las medidas de potencia activa se toman en los extremos de cada línea. Tales medidas de *potencia activa* que circula desde el nudo k hacia el nudo l de la línea $k-l$ se denota $\hat{p}_{kl}$ y tiene un grado de precisión indicado por σ_{kl}^p. Del mismo modo, las medidas de *potencia reactiva* se encuentran en ambos extremos de toda línea y se indica por $\hat{q}_{kl}$ con un grado de precisión asociado al parámetro σ_{kl}^q.

Además de la potencia activa, la potencia reactiva también viaja por la líneas de potencia. La potencia reactiva es una variable relacionada con los valores del voltaje. Si se dispone de suficiente potencia reactiva, el perfil del voltaje es adecuado, pero si no, tal perfil se deteriora. Finalmente, si la potencia reactiva disponible es mayor de la deseada, el perfil del voltaje se sobredimensiona.

El estado de la red de potencia se determina por los voltajes en los nodos. El valor del voltaje en cada nodo es un número complejo, expresado normalmente en forma polar. El módulo de este número complejo proporciona el valor del voltaje mientras que el ángulo es una medida de la "altura" relativa de este nudo. Cuanto mayor es la altura de un nudo, mayor es el flujo de la potencia activa desde este nudo hacia otros nudos conectados con él.

Para conocer el estado de la red de potencia, es necesario conocer las variables de estado. Típicamente se dispone de un número de medidas mayor que el mínimo requerido. Pero tales medidas no son exactas. Un modo racional de proceder consiste en usar todas las medidas para generar la mejor estimación del estado de la red.

Los voltajes (variables de estado) en la red se denotan por $v_i \angle \delta_i \quad i \in \Omega$, donde v_i es la magnitud del voltaje; δ_i es el ángulo del voltaje; y Ω, el conjunto de nudos de la red. Debe observarse que el ángulo de cualquier nudo fijo puede tomarse como el origen; así se pondrá $\delta_m = 0$, donde m es un nudo arbitrario.

Los ingenieros eléctricos han encontrado que la potencia activa desde el nudo k al l puede expresarse como

$$p_{kl}(v_k, v_l, \delta_k, \delta_l) = \frac{v_k^2}{z_{kl}} \cos\theta_{kl} - \frac{v_k v_l}{z_{kl}} \cos(\theta_{kl} + \delta_k - \delta_l) \quad (3.5)$$

donde p_{kl} es la potencia activa desde el nudo k al l, y $z_{kl}\angle\theta_{kl}$ es un número complejo relacionado con los parámetros fijos de la línea $k - l$.

Los ingenieros eléctricos también han encontrado que la potencia reactiva desde el nudo k al l a través de la línea $k - l$ puede expresarse como

$$q_{kl}(v_k, v_l, \delta_k, \delta_l) = \frac{v_k^2}{z_{kl}} \sin\theta_{kl} - \frac{v_k v_l}{z_{kl}} \sin(\theta_{kl} + \delta_k - \delta_l). \quad (3.6)$$

Los elementos principales de este problema son

1. **Datos.**

   $\hat{v}_i$: la magnitud medida del voltaje en el nudo i

   σ_i^v: la calidad de la medida del voltaje

   $\hat{p}_{kl}$: la medida de la potencia activa desde el nudo k hacia el nudo l de la línea $k - l$

   $\hat{q}_{kl}$: la medida de la potencia reactiva desde el nudo k hacia el nudo l de la línea $k - l$

   σ_{kl}^p: el grado de precisión de $\hat{p}_{kl}$

   σ_{kl}^q: el grado de precisión de $\hat{q}_{kl}$

   Ω: el conjunto de nudos de la red

   Ω_k: el conjunto de nudos conectados al nudo k

   $z_{k\ell}$: la magnitud de la impedancia asociada a la línea $k - l$

   $\theta_{k\ell}$: el ángulo de la impedancia asociado a la línea $k - l$

2. **Variables**

   v_i: la magnitud del voltaje en el nudo i

   δ_i: el ángulo del voltaje en el nudo i

3. **Restricciones:** En este caso no existen restricciones; esto es, el problema descrito es un problema de programación no lineal sin restricciones.

4. **Función a minimizar.** Para estimar el estado de una red de potencia (es decir, para determinar el valor de las variables de estado) se minimiza el error cuadrático de cada medida con respecto a su estimación, es decir, se minimiza

$$\sum_{i\in\Omega} \tfrac{1}{\sigma_i^v}(v_i - \hat{v}_i)^2 + \sum_{k\in\Omega, l\in\Omega_k} \tfrac{1}{\sigma_{kl}^p}(p_{kl}(v_k, v_l, \delta_k, \delta_l) - \hat{p}_{kl})^2$$
$$+ \sum_{k\in\Omega, l\in\Omega_k} \tfrac{1}{\sigma_{kl}^q}(q_{kl}(v_k, v_l, \delta_k, \delta_l) - \hat{q}_{kl})^2$$

donde Ω_k es el conjunto de nudos conectados al nudo k, y $p_{kl}(v_k, v_l, \delta_k, \delta_l)$ y $q_{kl}(v_k, v_l, \delta_k, \delta_l)$ están dados en (3.5) y (3.6). Obsérvese cómo los factores $1/\sigma_i^v$, $1/\sigma_{kl}^p$, y $1/\sigma_{kl}^q$ ponderan los errores según las calidades de las mediciones.

Ejemplo 3.1 (estimación del estado de una red). Considérese un circuito de dos nudos y una única línea conectándolos. La línea se caracteriza por la constante $z_{12}\angle\theta_{12} = 0.15\angle 90°$. Las magnitudes de las mediciones de los voltajes en los nudos 1 y 2 son respectivamente 1.07 y 1.01. Las mediciones de potencia activa en ambos extremos de la línea son respectivamente 0.83 y 0.81, y las medidas de potencia reactiva en ambos extremos son respectivamente 0.73 y 0.58.

El origen para los ángulos se toma en el nudo 2. Todos los aparatos medidores tiene idéntica precisión.

El problema de la estimación del estado de dicho circuito se formula minimizando

$$
\begin{aligned}
Z = \quad & (v_1 - 1.07)^2 + (v_2 - 1.01)^2 + (\tfrac{1}{0.15} v_1 v_2 \sin\delta_1 - 0.83)^2 \\
& +(-\tfrac{1}{0.15} v_1 v_2 \sin\delta_1 - 0.81)^2 + (\tfrac{1}{0.15} v_1^2 - \tfrac{1}{0.15} v_1 v_2 \cos\delta_1 - 0.73)^2 \\
& +(\tfrac{1}{0.15} v_2^2 - \tfrac{1}{0.15} v_1 v_2 \cos\delta_1 - 0.58)^2
\end{aligned}
$$

sin restricción alguna.

La solución de este problema es

$$
\begin{aligned}
v_1 &= 1.045 \\
v_2 &= 1.033 \\
\delta_1 &= 0.002
\end{aligned}
$$

■

3.4.2 Reparto óptimo de carga

El propósito de una red de trasmisión de potencia es transportar la potencia eléctrica desde los generadores hasta los puntos de demanda. El objetivo del problema del flujo de potencia óptima consiste en determinar la producción de potencia de cada generador de modo que toda la demanda se satisface con coste mínimo al tiempo que se respetan las restricciones propias de la red. Además de satisfacer la demanda, los valores del voltaje a lo largo de la red debe mantenerse en unos niveles aceptables. La potencia reactiva debe transmitirse a lo largo de la red, y su demanda debe ser satisfecha.

La potencia activa neta (generación menos demanda) que llega a un nudo debe expresarse como función de todos los voltajes y ángulos en la red

$$
p_{Gi} - P_{Di} = v_i \sum_{k=1}^{n} y_{ik} v_k \cos(\delta_i - \delta_k - \theta_{ik})
$$

donde p_{Gi} es la potencia activa generada en el nudo i; P_{Di}, la potencia activa demandada en el nudo i; v_i, la magnitud del voltaje; δ_i, el ángulo en el nudo i; y_{ik}, el módulo; y θ_{ik}, el argumento de una constante compleja que depende de la topología y la estructura física de la red; n, el número de nudos de la red.

De manera análoga, la potencia reactiva neta que llega a un nudo i puede expresarse en función de todas las magnitudes y ángulos del voltaje en la red;

$$q_{Gi} - Q_{Di} = v_i \sum_{k=1}^{n} y_{ik} v_k \sin(\delta_i - \delta_k - \theta_{ik})$$

donde q_{Gi} es la potencia reactiva generada en el nudo i; y Q_{Di} es la potencia reactiva demandada en el nudo i.

La magnitud del voltaje de todo nudo debe estar limitada superior e inferiormente

$$\underline{V}_i \leq v_i \leq \overline{V}_i$$

donde $\underline{V}_i$ es la cota inferior para la magnitud del voltaje en el nudo i, y $\overline{V}_i$ la cota superior.

Los generadores pueden producir potencia activa por encima de una cierta cota inferior y por debajo de una cierta cota superior

$$\underline{P}_{Gi} \leq p_{Gi} \leq \overline{P}_{Gi}$$

donde $\underline{P}_i$ es la mínima potencia activa que puede salir del generador i y $\overline{P}_i$ es la máxima.

De la misma manera, los generadores pueden producir potencia reactiva entre una cota inferior y una superior

$$\underline{Q}_{Gi} \leq q_{Gi} \leq \overline{Q}_{Gi}$$

donde $\underline{Q}_i$ es la mínima potencia reactiva de salida en el generador i y $\overline{Q}_i$ es la máxima.

Los cuatro elementos de este problema son

1. **Datos**

n: el número de nudos en la red

(y_{ik}, θ_{ik}): un número complejo que tiene módulo y_{ik}, y argumento θ_{ik} el cual depende de la topología y estructura física de la red

P_{Di}: la demanda de potencia activa en el nudo i

Q_{Di}: la demanda de potencia reactiva en el nudo i

$\underline{V}_i$: la cota inferior para el módulo del voltaje en el nudo i

$\overline{V}_i$: la cota superior para el módulo del voltaje en el nudo i

$\underline{P}_{Gi}$: la potencia activa de salida mínima del generador i

$\overline{P}_{Gi}$: la potencia activa de salida máxima del generador i

$\underline{Q}_{Gi}$: la potencia reactiva de salida mínima del generador i

$\overline{Q}_{Gi}$: la potencia reactiva de salida máxima del generador i

C_i: el precio por unidad de potencia activa en el generador i

2. **Variables**

   v_i: el voltaje en el nudo i

   δ_i: el ángulo en el nudo i

   p_{Gi}: la potencia activa generada en el nudo i

   q_{Gi}: la potencia reactiva generada en el nudo i

3. **Restricciones.** Hay distintos tipos de condiciones.

   (a) Equilibrio de potencia activa:

   $$p_{Gi} - P_{Di} = v_i \sum_{k=1}^{n} y_{ik} v_k \cos(\delta_i - \delta_k - \theta_{ik}), \quad \forall i = 1, 2, \ldots, n$$

   (b) Equilibrio de potencia reactiva:

   $$q_{Gi} - Q_{Di} = v_i \sum_{k=1}^{n} y_{ik} v_k \sin(\delta_i - \delta_k - \theta_{ik}), \quad \forall i = 1, 2, \ldots, n$$

   (c) Cotas para las variables:

   $$\begin{array}{rcccll} \underline{V}_i & \leq & v_i & \leq & \overline{V}_i, & \forall i = 1, 2, \ldots, n \\ \underline{P}_{Gi} & \leq & p_{Gi} & \leq & \overline{P}_{Gi}, & \forall i = 1, 2, \ldots, n \\ \underline{Q}_{Gi} & \leq & q_{Gi} & \leq & \overline{Q}_{Gi}, & \forall i = 1, 2, \ldots, n \end{array}$$

   (d) Cotas para los ángulos:

   $$-\pi \leq \delta_i \leq \pi, \quad \forall i = 1, 2, \ldots, n$$

   El origen para los ángulos se sitúa de modo arbitrario en cualquier nudo (por ejemplo, el nudo k, $\delta_k = 0$).

4. **Función a minimizar.** Dada C_i, el precio de producción de la unidad de potencia activa en el generador i, el flujo óptimo de potencia se convierte en la minimización de

   $$Z = \sum_{i=1}^{n} C_i p_{Gi}$$

   bajo todas las restricciones anteriores.

Ejemplo 3.2 (flujo óptimo de potencia). Considérese una red con 3 nudos y 3 líneas. Los parámetros de tal red son

$$\begin{array}{ll}
y_{11} = 22.97, & \theta_{11} = -1.338 \\
y_{22} = 21.93, & \theta_{22} = -1.347 \\
y_{33} = 20.65, & \theta_{33} = -1.362 \\
y_{12} = y_{21} = 12.13, & \theta_{12} = \theta_{21} = 1.816 \\
y_{13} = y_{31} = 10.85, & \theta_{13} = \theta_{31} = 1.789 \\
y_{23} = y_{32} = 9.81, & \theta_{23} = \theta_{32} = 1.768
\end{array}$$

El nudo 1 es un generador con cotas inferior y superior para la potencia activa generada de 0 y 3, respectivamente, y para la potencia reactiva de -1 y 2, respectivamente. Las cotas para la generación de potencia activa para el generador 2 son, respectivamente, 0 y 3, y para la potencia reactiva -1 y 2, respectivamente. El nudo 3 es de consumo con una demanda de 4.5 y 1.5. Los límites permitidos para el módulo del voltaje en los nudos 2 y 3 son, respectivamente, 0.95 y 1.10, y para el nudo 1 son 0.95 y 1.13, respectivamente. El precio de producción en el generador en el nudo 1 es 6 y en el generador del nudo 2 es 7. Supóngase un periodo de 1 hora y póngase el origen de voltajes en el nudo 3. Este problema tiene la siguiente estructura. Minimizar

$$6p_{G1} + 7p_{G2}$$

sujeto a

$$\begin{array}{rcl}
0 & = & p_{G1} - 22.97v_1^2\cos(1.338) - 12.13v_1v_2\cos(\delta_1 - \delta_2 - 12.127) \\
 & & -10.85v_1v_3\cos(\delta_1 - \delta_3 - 10.846) \\
0 & = & p_{G2} - 21.93v_2^2\cos(1.347) - 12.13v_2v_1\cos(\delta_2 - \delta_1 - 12.127) \\
 & & -9.81v_2v_3\cos(\delta_2 - \delta_3 - 9.806) \\
0 & = & -4.5 - 20.65v_3^2\cos(1.362) - 10.85v_3v_1\cos(\delta_3 - \delta_1 - 10.846) \\
 & & -9.81v_3v_2\cos(\delta_3 - \delta_2 - 9.806) \\
0 & = & q_{G1} - 22.97v_1^2\sin(1.338) - 12.13v_1v_2\sin(\delta_1 - \delta_2 - 12.127) \\
 & & -10.85v_1v_3\sin(\delta_1 - \delta_3 - 10.846) \\
0 & = & q_{G2} - 21.93v_2^2\sin(1.347) - 12.13v_2v_1\sin(\delta_2 - \delta_1 - 12.127) \\
 & & -9.81v_2v_3\sin(\delta_2 - \delta_3 - 9.806) \\
0 & = & -1.5 - 20.65v_3^2\sin(1.362) - 10.85v_3v_1\sin(\delta_3 - \delta_1 - 10.846) \\
 & & -9.81v_3v_2\sin(\delta_3 - \delta_2 - 9.806)
\end{array}$$

Tabla 3.1: Ejemplo de una tabla de contingencia

			t_{ij}		
	B_1	B_2	$\ldots$	B_n	r_i
A_1	t_{11}	t_{12}	$\ldots$	t_{1n}	r_1
A_2	t_{21}	t_{22}	$\ldots$	t_{2n}	r_2
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
A_m	t_{m1}	t_{m2}	$\ldots$	t_{mn}	r_m
c_j	c_1	c_2	$\ldots$	c_n	

$$
\begin{array}{lccrr}
0.95 & \leq & v_1 & \leq & 1.13 \\
0.95 & \leq & v_2 & \leq & 1.10 \\
0.95 & \leq & v_3 & \leq & 1.10 \\
0 & \leq & p_{G1} & \leq & 3 \\
0 & \leq & p_{G2} & \leq & 3 \\
-1 & \leq & q_{G1} & \leq & 2 \\
-1 & \leq & q_{G2} & \leq & 2 \\
-\pi & \leq & \delta_1 & \leq & \pi \\
-\pi & \leq & \delta_2 & \leq & \pi \\
 & & \delta_3 & = & 0
\end{array}
$$

La solución óptima local, que se ha obtenido mediante GAMS (véase la Sección 11.4.11), es

$$
\begin{array}{lcl}
Z & = & 30.312 \\
p_{G1} & = & 3.000 \\
p_{G2} & = & 1.759 \\
q_{G1} & = & 1.860 \\
q_{G2} & = & 0.746 \\
v_1 & = & 1.130 \\
v_2 & = & 1.100 \\
v_3 & = & 0.979 \\
\delta_1 & = & 0.190 \\
\delta_2 & = & 0.174
\end{array}
$$

■

3.5 El problema de la matriz equilibrada

Los ingenieros, estadísticos, sociólogos, y otros, usan información experimental que se estructura en tablas de contingencia (véase la tabla 3.1).

Los elementos en estas tablas se clasifican de acuerdo con dos criterios. El primero, llamado A, tiene m categorías y el segundo, llamado B, tiene n categorías. De este modo, cada elemento puede clasificarse en una de las $m \times n$ celdas.

El contenido de estas celdas puede variar en el tiempo. El problema de la matriz equilibrada está relacionado con la actualización de estas tablas, suponiendo que se conocen las sumas en filas y columnas de las tablas actualizadas.

Por ejemplo, imagínese que nos interesa conocer la inmigración y cómo ésta cambia con el tiempo. Cada flujo migratorio debe clasificarse según su origen y destino. El objetivo consiste en estimar el número de personas que emigra desde cada par de ciudades. Para este fin, se puede trabajar con un censo que, debido a razones de costes, se hace cada 10 años. Además se dispone de información anual sobre las emigraciones netas en cada ciudad, y que debe usarse para actualizar las tablas de contingencia anuales.

Supóngase una tabla de contingencia desactualizada $m \times n$ con entradas $\{t_{ij}\}$, y denótese la tabla actualizada como $\mathbf{T} = (T_{ij})$, donde T_{ij} es el contenido de la celda $i-j$. Supóngase que se conocen las sumas de las columnas y las filas de la nueva matriz $\mathbf{T}$, es decir, se debe respetar la restricción

$$\sum_{j=1}^{n} T_{ij} = r_i, i = 1, \ldots, m \tag{3.7}$$

$$\sum_{i=1}^{m} T_{ij} = c_j, j = 1, \ldots, n \tag{3.8}$$

donde los parámetros $r_i; i = 1, \ldots, m$ y $c_j; j = 1, \ldots, n$ son conocidos.

Se necesitan introducir condiciones adicionales que reflejen que (en la mayoría de las aplicaciones) la celda representa contadores de elementos (son no negativos):

$$T_{ij} \geq 0, \; i = 1, \ldots, m; \;\; j = 1, \ldots, n$$

Nuestro objetivo consiste en obtener una matriz con la misma estructura que la anterior pero verificando las restricciones en la suma de filas y columnas. Así, la función objetivo seleccionada es una distancia entre $\{t_{ij}\}$ y $\{T_{ij}\}$:

$$Z = F(\mathbf{T}, \mathbf{t}) \tag{3.9}$$

Una posibilidad consiste en usar la función ponderada de mínimos cuadrados

$$Z = \sum_{i=1}^{m} \sum_{j=1}^{n} \omega_{ij}(T_{ij} - t_{ij})^2 \tag{3.10}$$

donde ω_{ij} (los pesos) son escalares positivos dados. Otra alternativa es la función de entropía (cambiada de signo)

$$Z = \sum_{i=1}^{m} \sum_{j=1}^{n} \frac{T_{ij} \log T_{ij}}{t_{ij}} - T_{ij} + t_{ij} \tag{3.11}$$

Obsérvese cómo $Z = 0$ cuando $T_{ij} = t_{ij}; \forall i, j$, y que Z aumenta con la discrepancia entre T_{ij} y t_{ij}.

Resumiendo, los elementos de este problema son

Tabla 3.2: Matriz origen-destino obtenida como resultado de la encuesta

	t_{ij}				
	1	2	3	4	r_i
1	–	60	275	571	906
2	50	–	410	443	903
3	123	61	–	47	231
4	205	265	75	–	545
c_j	378	386	760	1061	2585

1. **Datos**

   r_i: la suma de todos los elementos de la fila i

   c_j: la suma de todos los elementos de la columna j

   t_{ij}: el contenido observado de la celda $i-j$

2. **Variables**

   T_{ij}: el contenido estimado de la celda $i-j$

3. **Restricciones**

$$\sum_{j=1}^{n} T_{ij} = r_i, i = 1, \ldots, m \tag{3.12}$$

$$\sum_{i=1}^{m} T_{ij} = c_j, j = 1, \ldots, n \tag{3.13}$$

$$T_{ij} \geq 0, \; i = 1, \ldots, m; \; j = 1, \ldots, n \tag{3.14}$$

4. **Función a minimizar.** La función (3.10) ó (3.11).

Ejemplo 3.3 (distribución de viajes). Un ejemplo de un problema práctico que puede abordarse en el contexto de este modelo es la predicción de la matriz de distribución T de viajes en una ciudad. En este ejemplo se trata esta situación. Considérese un área pequeña que se ha dividido en cuatro zonas y supóngase que una cierta encuesta ha dado como fruto la matriz de viajes de la tabla 3.2.

Los dos conjuntos de restricciones (3.7)–(3.8) reflejan nuestro conocimiento sobre la generación y atracción de viajes en las zonas del área bajo estudio. Nos interesan las entradas de la matriz que pueden interpretarse como viajes. Las estimaciones del número total de viajes futuros que acaban o salen de cada zona se dan en la tabla 3.3.

Tabla 3.3: Estimaciones del número total de viajes futuros en cada zona

Zonas	Destinos futuros estimados (r_i)	Orígenes futuros estimados (c_j)
1	12,000	6,750
2	10,500	7,300
3	3,800	10,000
4	7,700	9,950

Tabla 3.4: Solución basada en la función de entropía

	T_{ij}				
	1	2	3	4	r_i
1	–	1997.893	4176.466	5825.642	12,000
2	1364.381	–	5293.379	3842.240	10,500
3	2322.858	1195.024	–	282.118	3,800
4	3062.761	4107.084	530.156	–	7,700
c_j	6750	7300	10,000	9950	34,000

Tabla 3.5: Solución basada en mínimos cuadrados ponderados

	T_{ij}				
	1	2	3	4	r_i
1	–	1700.324	4253.616	6046.060	12,000
2	1250.923	–	5484.550	3764.527	10,500
3	2396.886	1263.701	–	139.413	3,800
4	3102.191	4335.975	261.835	–	7,700
c_j	6750	7300	10,000	9950	34,000

En este problema las variables intrazonales se han eliminado, es decir, las variables T_{ii} $i = 1, \ldots, 4$ han sido eliminadas de la formulación.

Se han usado dos contrastes. El primero se basa en (3.11), y la solución se da en la tabla 3.4. Las estimaciones en el segundo caso se han obtenido mediante (3.9), con $\omega_i = 1/T_{ij}$. La solución se encuentra en la tabla 3.5.

∎

3.6 El problema de la asignación de tráfico

La planificación del tráfico ha motivado un buen número de modelos matemáticos. El análisis de estos modelos ayuda a planificar y predecir los efectos que determinados cambios en la red de tráfico tendrán en la buena marcha de la red. El esquema típico de planificación del transporte usado en las aplicaciones consta de cuatro etapas:

1. **Fase de generación de viajes.** Esta etapa comienza considerando un sistema de zonificación y una serie de datos relativos a cada zona del estudio. Estos datos, que incluyen información sobre la actividad económica, la distribución social, los recursos educacionales y lúdicos, los espacios para compras, se usan para estimar el número total de viajes generados y atraídos a cada zona bajo estudio.

2. **Fase de distribución.** La siguiente etapa consiste en la adjudicación de estos viajes entre orígenes y destinos, determinando la llamada matriz de viajes origen-destino.

3. **Descomposición modal.** A continuación, la descomposición modal produce la adjudicación de viajes a modos diversos. En esta fase las matrices origen–destino se obtienen para cada modo de transporte. Sus elementos son el número total de viajes por modo de transporte para cada par origen–destino O–D ω.

4. **Asignación.** Finalmente, la última etapa requiere la *asignación* de estos viajes a la red de transporte. Este ejemplo trata sobre la asignación de automóviles privados a la red de tráfico.

Debe introducirse un principio que gobierne el comportamiento de los usuarios al elegir la ruta en la red. Wardrop [102]) fué el primero en enunciar formalmente este principio: "Bajo condiciones de equilibrio, el tráfico se organiza en redes congestionadas de tal modo que ningún vehículo puede reducir el tiempo de viaje mediante un cambio de ruta."

Este principio se ha usado como punto de partida para confeccionar modelos de asignación en equilibrio. Un corolario de este principio es que si todos los viajeros perciben el tiempo de los viajes del mismo modo, bajo condiciones de equilibrio, todas las rutas utilizadas entre un par O-D tienen el mismo tiempo mínimo mientras las no usadas requieren un tiempo igual o mayor.

Beckman et al. [10] formularon el siguiente problema de optimización para expresar las condiciones de equilibrio que se derivan del primer principio de Wardrop. Este modelo predice el nivel de uso de los diferentes arcos de la red. Así, puede usarse para responder cuestiones como qué ocurriría en el nivel de uso de la red si se construyera una nueva carretera o si la capacidad de una determinada ruta se modificara.

Los elementos principales de este problema son:

1. **Datos**

$(\mathcal{N}, \mathcal{A})$: una grafo dirigido $(\mathcal{N}, \mathcal{A})$, que se entiende como un modelo de la red de tráfico con un conjunto de nodos $\mathcal{N}$, y un conjunto de arcos $\mathcal{A}$ que representan las calles. El conjunto de nodos $\mathcal{N}$ del grafo representan intersecciones o también los llamados centroides, que indican las zonas del estudio (orígenes y destinos).

W: el conjunto de pares orígenes–destinos.

d_ω: datos de entrada que representan el número de viajes en coche desde el origen i al destino j, para cada par origen–destino $\omega = (i, j)$. La matriz de pares origen–destino $\{d_\omega\}_{\omega \in W}$ se obtiene en la fase de distribución modal.

$C_a(f_a)$: una función de coste que indica el retraso en el arco $a \in \mathcal{A}$, para cada arco $(i, j) \in \mathcal{A}$, como función del flujo total f_a que lleva el mismo arco a.

$\mathcal{R}_\omega$: el conjunto de rutas para el par $\omega = (i, j)$.

2. **Variables**

h_r: el flujo en la ruta r

f_a: el flujo en el arco a

3. **Restricciones.** El número de usuarios de un par origen–destino ω es la suma del número total de usuarios en caminos distintos que satisfacen tal par:

$$\sum_{r \in \mathcal{R}_\omega} h_r = d_\omega, \ \forall \omega \in W \tag{3.15}$$

Además, el flujo de cada camino debe ser no negativo:

$$h_r \geq 0, \ \forall r \in \mathcal{R}_\omega, \ \forall \omega \in W \tag{3.16}$$

El flujo de cada arco a es la suma del flujo en todos los caminos que lo usan:

$$\sum_{w \in W} \sum_{r \in \mathcal{R}_\omega} \delta_{a,r} h_r = f_a \quad \forall a \in \mathcal{A}$$

donde $\delta_{a,r} = 1$ si $r \in \mathcal{R}_\omega$ contiene el arco a, y 0 en otro caso.

4. **Función a optimizar.** El problema de asignación de tráfico minimiza la siguiente función:

$$Z = \sum_{a \in \mathcal{A}} \int_0^{f_a} C_a(x) dx \tag{3.17}$$

Como resultado del volumen creciente de tráfico, la velocidad en los arcos tiende a disminuir. La función C_a, es decir el tiempo necesario para atravesar el arco a, tiene en cuenta este hecho. Estas funciones en el análisis de sistemas

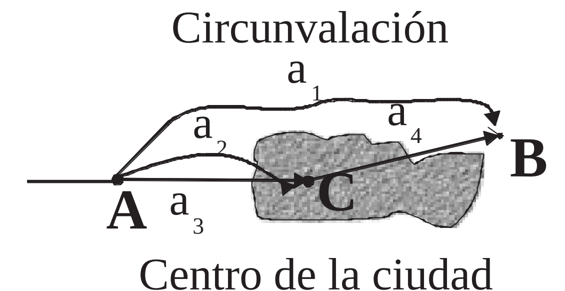


Figura 3.8: Diagrama para la red de carreteras.

de tráfico son positivas, no lineales y estrictamente crecientes. Los parámetros que relacionan el tiempo de viaje, C_a, en el arco en función del flujo f_a, en él, es el *tiempo de viaje libre de flujo*, c_a^0, y la capacidad práctica del arco, k_a, que es una medida del flujo a partir del cual, el tiempo de viaje se incrementará muy rápidamente si el flujo aumenta. La expresión más común para $C_a(f_a)$, llamada la función BPR, es

$$C_a(f_a) = c_a^0 + b_a \left(\frac{f_a}{k_a}\right)^{n_a} \tag{3.18}$$

Ejemplo 3.4 (problema de asignación de tráfico). Considérese el problema de una ciudad con una vía de circunvalación y varias rutas centrales según se ilustra en la Figura 3.8. Imagínese que se realizan 4000 viajes desde A hasta B, y 2500 desde A hasta C. Las rutas disponibles para satisfacer la demanda del par $\omega_1 = (A, B)$ son $r_1 = a_1$, $r_2 = a_2 - a_4$, y $r_3 = a_3 - a_4$, y las rutas para el par $\omega_2 = (A, C)$ son $r_4 = a_2$ y $r_5 = a_3$. En este ejemplo $W = \{w_1, w_2\}$, y $\mathcal{R}_{\omega_1} = \{r_1, r_2, r_3\}$ y $\mathcal{R}_{\omega_2} = \{r_4, r_5\}$. Las variables de flujo en los caminos son $h_1, \ldots, h_5$, y las variables de flujo en los arcos son $f_1, \ldots, f_4$.

Como

$$\begin{aligned}\int_0^{f_a} C_a(x)dx &= \int_0^{f_a} c_a^0 + b_a \left(\frac{x}{k_a}\right)^{n_a} dx \\ &= c_a^0 f_a + \frac{b_a}{n_a+1}\left(\frac{f_a}{k_a}\right)^{n_a+1}\end{aligned}$$

la formulación completa es como sigue. Minimizar

$$Z = \sum_{i=1}^{4} \int_0^{f_{a_i}} C_{a_i}(x)dx = \sum_{a \in \mathcal{A}} c_{a_i}^0 f_{a_i} + \frac{b_{a_i}}{n_{a_i}+1}\left(\frac{f_{a_i}}{k_{a_i}}\right)^{n_{a_i}+1} \tag{3.19}$$

sujeto a

$$h_1 + h_2 + h_3 = 4000 \tag{3.20}$$
$$h_4 + h_5 = 2500 \tag{3.21}$$
$$h_1 = f_1 \tag{3.22}$$

Tabla 3.6: .Parámetros para las funciones BPR

Enlace a	k_a	c_a^0	b_a	n_a
1	500	5	1	4
2	400	7	1	4
3	400	10	1	4
4	500	2	1	4

Tabla 3.7: Estimación de los viajes

Flujo en los arcos		Flujo en las rutas	
a_1	3845.913	r_1	3845.913
a_2	2354.419	r_2	154.087
a_3	299.667	r_3	0.000
a_4	154.087	r_4	2200.333
		r_5	299.667

$$h_2 + h_4 = f_2 \quad (3.23)$$
$$h_3 + h_5 = f_3 \quad (3.24)$$
$$h_2 + h_3 = f_4 \quad (3.25)$$
$$h_1, \ldots, h_5 \geq 0 \quad (3.26)$$

En este ejemplo se han usado las funciones en (3.18), y la tabla 3.6 muestra los parámetros. La solución correspondiente se encuentra en la tabla 3.7.

■

Ejercicios

3.1 Una mesa debe pasar a través de un ángulo recto en un pasillo de A unidades de anchura hacia otro de B unidades de ancho. ¿Cuáles son los tamaños de la mesa (a unidades de ancho y b unidades de largo) que permite tal movimiento?

3.2 Se desea construir un hospital entre dos ciudades que distan D entre sí. El lugar elegido corresponde al menos contaminado en la línea que une las dos ciudades. Sabiendo que la contaminación es proporcional a los residuos de las industrias presentes y al inverso de la distancia más 1, y que la segunda ciudad genera 3 veces más actividad industrial que la primera, determinar el lugar óptimo para el hospital.

3.3 Encontrar la mínima longitud de una escalera que debe apoyarse sobre la pared si una caja de dimensiones a y b está colocada justo en el rincón de

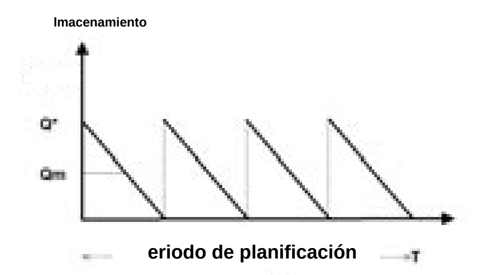


Figura 3.9: Evolución de la reserva con el tiempo.

Tabla 3.8: Coste del almacenamiento y del pedido de los dos artículos

Bien	C_s	C_r	Demanda
A	1 dólar	18 dólares	365 toneladas
B	2 dólares	25 dólares	365 toneladas

esa misma pared. Formular el problema como una situación de programación no lineal.

3.4 Se desea encontrar la máxima longitud de una viga que debe moverse a través de la puerta de una sala suponiendo que la altura de la puerta es h y está situada una distancia d de la pared frontal. La altura de la sala puede suponerse infinita. Formular esta situación como un problema de programación no lineal.

3.5 Considérese la siguiente situación de gestión de inventarios. La demanda de un cierto bien es constante a lo largo de un horizonte de 365 días. La persona encargada del cuidado del bien tiene en cuenta un coste de almacenamiento de $C_s = 1$ dólar por día y tonelada del citado bien, y un precio fijo para preparar un pedido (con independencia del tamaño) de $C_r = 18$ dólares. La demanda es 365 toneladas para el horizonte considerado. La Figura 3.9 indica la evolución de la reserva asumiendo que el almacén está vacío al principio del periodo, y se ha realizado una petición en el instante inicial. La reserva media, Q_m, se calcula como $Q/2$, donde Q es la cantidad pedida. Formular un problema de programación no lineal para decidir el tamaño del pedido con objeto de minimizar el coste de la gestión del inventario.

3.6 Repítase el problema 3.5 añadiendo un segundo producto. Los parámetros para cada artículo están en la tabla 3.8. Supóngase que el almacén tiene un capacidad máxima de 60 toneladas. Formular el correspondiente problema como una situación de programación no lineal.

3.7 Un productor de electricidad debe planificar su producción de energía eléctrica cada hora para maximizar los beneficios durante un determina-

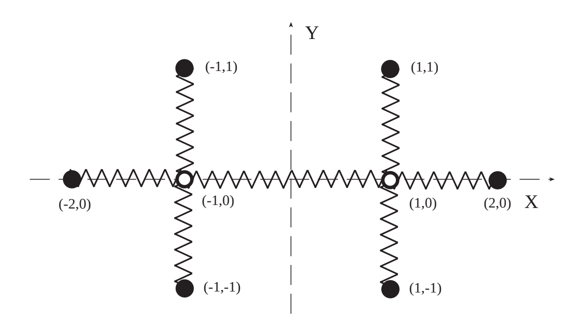


Figura 3.10: Sistema de muelles.

do número de horas. Formular el problema de programación no lineal subyacente teniendo en cuenta que:

(a) El productor no trabaja antes del periodo citado.

(b) Los precios por hora de la energía decrecen linealmente con la producción de energía en la hora correspondiente.

(c) La energía mínima que puede producirse cada hora es cero y la máxima una determinada cantidad.

(d) La producción de energía en dos horas consecutivas no puede diferir en más de una cantidad preestablecida.

(e) El precio de producción es lineal.

3.8 Examínese el sistema de muelles de la Figura 3.10. Los puntos negros están fijos mientras que los blancos corresponden a enlaces libres. Todos los nodos pueden rotar libremente de modo que los muelles siempre se sitúan a lo largo de la línea que une los dos extremos. Cada muelle se caracteriza por una constante positiva k_i, $i = 1, \ldots, 7$ en unidades apropiadas. Se postula que la configuración de equilibrio de los puntos blancos es la que minimiza la energía global de todos los muelles donde la energía de cada muelle es proporcional (con constante k_i) al cuadrado de la distancia entre los extremos. Formular el problema como un problema de programación no lineal.

Parte II

Métodos

Capítulo 4

Introducción a la programación lineal

4.1 Introducción

La programación lineal tiene muchas aplicaciones de interés, como las ilustradas con los distintos ejemplos del capítulo 1, donde se introdujeron los elementos básicos de un PPL. En este capítulo se presenta una descripción más formal de la programación lineal. En concreto, se definen los conceptos más importantes relacionados con un PPL y se describen los diversos tipos de soluciones posibles. En la sección 4.2 se introduce el problema general de la programación lineal, y se ilustran los casos de solución única y múltiple, y los casos de solución no acotada y de infactibilidad. En la sección 4.3 se describe la forma estándar de un problema de programación lineal y se muestra cómo cualquier PPL puede ponerse en esa forma. La sección 4.4 define y analiza las soluciones básicas, que son las soluciones candidatas a lograr un valor óptimo de la función objetivo. En la sección 4.5 se analizan las sensibilidades del problema con respecto a los valores de los términos independientes de las restricciones. Finalmente, en la sección 4.6 se muestra que todo problema de programación lineal tiene un problema dual asociado, y se proporcionan varios ejemplos en los que se interpretan ambos problemas, primal y dual.

4.2 Formulación del problema

El objeto de la programación lineal es optimizar (minimizar o maximizar) una función lineal de n variables sujeto a restricciones lineales de igualdad o desigualdad. Más formalmente, se dice que un problema de programación lineal consiste en encontrar el óptimo (máximo o mínimo) de una función lineal en un conjunto que puede expresarse como la intersección de un número finito de hiperplanos y semiespacios en $\mathbb{R}^n$. Considérense las siguientes definiciones.

Definición 4.1 (problema de programación lineal). *La forma más general de un problema de programación lineal (PPL) cosiste en minimizar o maximizar*

$$Z = f(\mathbf{x}) = \sum_{j=1}^{n} c_j x_j \tag{4.1}$$

sujeto a

$$\begin{array}{rcll} \sum_{j=1}^{n} a_{ij}x_j & = & b_i, & i = 1, 2, \ldots p-1 \\ \sum_{j=1}^{n} a_{ij}x_j & \geq & b_i, & i = p, \ldots, q-1 \\ \sum_{j=1}^{n} a_{ij}x_j & \leq & b_i, & i = q, \ldots, m \end{array} \tag{4.2}$$

donde p, q, y m son enteros positivos tales que

$$1 \leq p \leq q \leq m$$

■

Lo que distingue un problema de programación lineal de cualquier otro problema de optimización es que todas las funciones que en él intervienen son lineales. Una única función no lineal hace que el problema no pueda clasificarse como problema de programación lineal. Téngase en cuenta además que en este libro se considerará que los problemas tienen siempre un número finito de restricciones. La función (lineal) en (4.1) se denomina función objetivo o función de coste, y es la función de ha de optimizarse. Obsérvese que en (4.2) se presentan todas las posibles alternativas en lo que se refiere a los operadores que relacionan los dos términos de las restricciones (lineales), dependiendo de los valores p y q. Como casos especiales, el problema puede tener exclusivamente restricciones de igualdad, de desigualdad de un tipo, de desigualdad del otro tipo, desigualdades de ambos tipos, igualdades y desigualdades, etc. Salvo que se indique lo contrario, en este libro se consideran problemas de minimización.

Definición 4.2 (solución factible). *Un punto $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ que satisface todas las restricciones (4.2) se denomina solución factible. El conjunto de todas esas soluciones es la región de factibilidad .* ■

Definición 4.3 (solución óptima). *Un punto factible $\tilde{\mathbf{x}}$ tal que $f(\mathbf{x}) \geq f(\tilde{\mathbf{x}})$ para cualquier otro punto factible $\mathbf{x}$ se denomina una solución óptima del problema.* ■

El objetivo de los problemas de optimización es encontrar un óptimo global. Sin embargo, las condiciones de optimalidad sólo garantizan, en general, óptimos locales, si éstos existen (véase el capítulo 8). Sin embargo, los problemas lineales presentan propiedades que hacen posible garantizar el óptimo global:

- Si la región factible está acotada, el problema siempre tiene una solución (ésta es una condición suficiente pero no necesaria para que exista una solución).
- El óptimo de un problema de programación lineal es siempre un óptimo global.
- Si $\mathbf{x}$ y $\mathbf{y}$ son soluciones óptimas de un problema de programación lineal, entonces cualquier combinación (lineal) convexa de lo mismos también es una solución óptima. Obsérvese que las combinaciones convexas de puntos con el mismo valor de la función de coste presentan el mismo valor de la función de coste.
- La solución óptima se alcanza siempre, al menos, en un punto extremo de la región factible. Esto se muestra en el Teorema 4.2, de la sección 4.2.

Los ejemplos siguientes ilustran problemas con solución única, múltiple, solución no acotada y solución infactible.

Ejemplo 4.1 (solución única). Considérese el siguiente problema de programación lineal. Maximizar

$$Z = 3x_1 + x_2$$

sujeto a

$$\begin{array}{rrrr} -x_1 & +x_2 & \leq & 2 \\ x_1 & +x_2 & \leq & 6 \\ x_1 & & \leq & 3 \\ 2x_1 & -x_2 & \leq & 4 \\ & -x_2 & \leq & 0 \\ -x_1 & -x_2 & \leq & -1 \\ -x_1 & & \leq & 0 \end{array} \tag{4.3}$$

La figura 4.1 muestra la región factible (área sombreada), las curvas de nivel (líneas finas) de la función objetivo, el vector gradiente indicando la dirección de crecimiento de la función objetivo. La solución se alcanza en el punto P, dado que se encuentra en la última curva de nivel en la dirección indicada y en el último punto factible de la región de factibilidad. P es por tanto la intersección de las rectas

$$\begin{array}{rrrr} x_1 & +x_2 & = & 6 \\ x_1 & & = & 3 \end{array}$$

Por tanto, el máximo, $Z = 12$, se alcanza en el punto

$$P = (3, 3)$$

■

Ejemplo 4.2 (soluciones múltiples). Si la función objetivo del Ejemplo 4.1 se cambia por la función

$$Z = x_1 + x_2$$

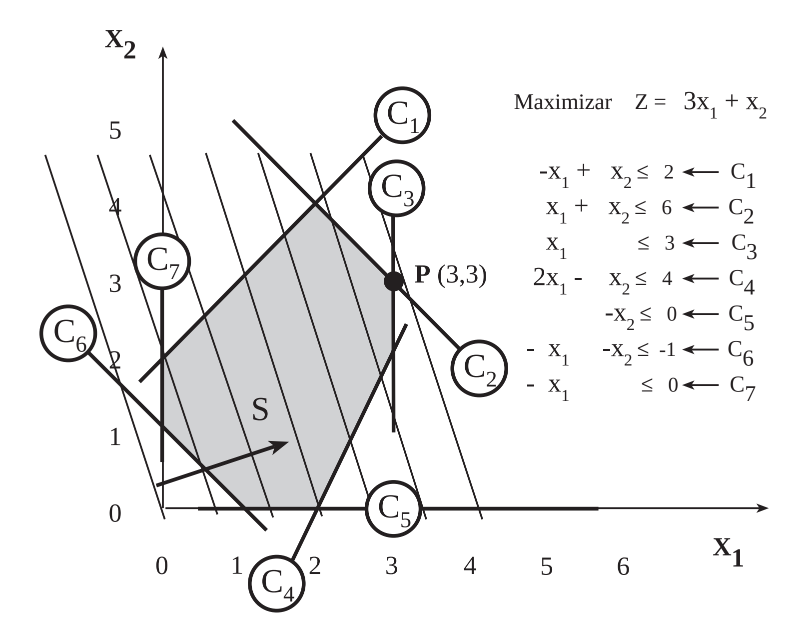


Figura 4.1: Ilustración gráfica de un problema de programación lineal con solución única.

las curvas de nivel resultantes son paralelas a una de las restricciones (la segunda). En este caso, el óptimo se alcanza en todos los puntos de la arista correspondiente del politopo, como se muestra en la figura 4.2. Cualquier punto del segmento de línea recta entre los puntos $(2,4)^T$ y $(3,3)^T$ resulta ser un máximo global del problema ($Z = 6$). ■

Ejemplo 4.3 (solución no acotada). Considérese el siguiente PPL en el que se maximiza

$$Z = 3x_1 + x_2 \tag{4.4}$$

sujeto a

$$\begin{array}{rrrr} -x_1 & +x_2 & \leq & 2 \\ & -x_2 & \leq & 0 \\ -x_1 & -x_2 & \leq & -1 \\ -x_1 & & \leq & 0 \end{array} \tag{4.5}$$

que tiene una solución no acotada, porque como se muestra en la figura 4.3, la región factible no está acotada en la dirección de crecimiento de la función objetivo. ■

Ejemplo 4.4 (solución infactible). Considérese el siguiente PPL: Maximizar

$$Z = 3x_1 + x_2 \tag{4.6}$$

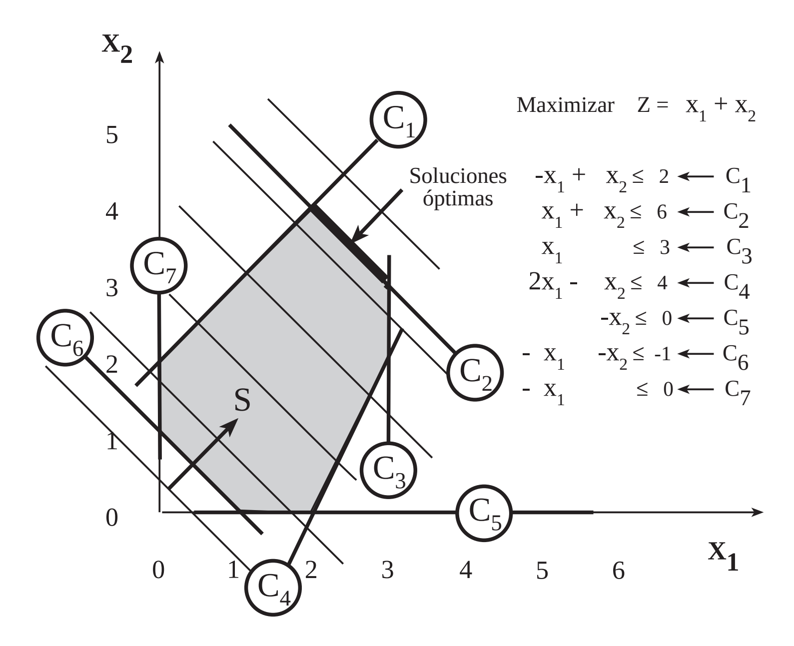


Figura 4.2: Ilustración gráfica de un problema de programación lineal con soluciones múltiples.

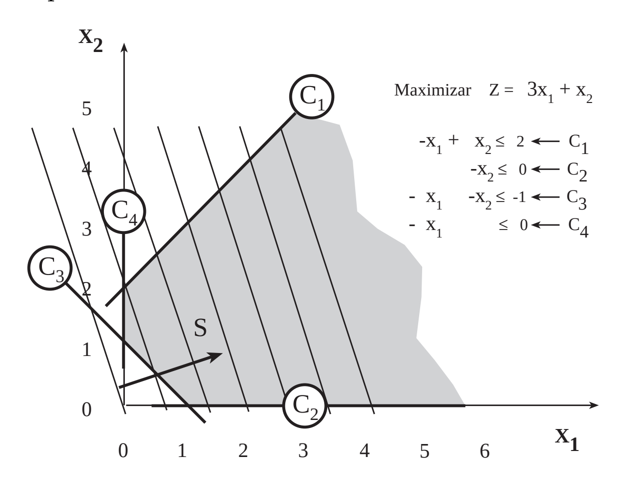


Figura 4.3: Ilustración gráfica de un problema de programación lineal con región de factibilidad y solución no acotada.

sujeto a

$$\begin{array}{rrcr} -x_1 & +x_2 & \leq & 2 \\ x_1 & +x_2 & \leq & 6 \\ x_1 & & \leq & 3 \\ 2x_1 & -x_2 & \leq & 4 \\ & -x_2 & \leq & 0 \\ -x_1 & -x_2 & \leq & -1 \\ -x_1 & & \leq & 0 \\ x_1 & +x_2 & \leq & 0 \end{array} \tag{4.7}$$

Este problema no es factible porque la nueva restricción $x_1 + x_2 \leq 0$ no es compatible con las restricciones previas. ■

Los vértices de los politopos (polígonos) en las figuras 4.1–4.3 se denominan puntos extremos.

4.3 Problema de programación lineal en forma estándar

Dado que un PPL puede plantearse de diversas formas, para unificar su análisis, es conveniente transformarlo en lo que normalmente se llama *forma estándar.* A veces, esta transformación ha de realizarse antes de resolver el PPL y determinar el óptimo. Para describir un PPL en forma estándar son necesarios los tres elementos siguientes:

1. Un vector $\mathbf{c} \in \mathbb{R}^n$
2. Un vector no negativo $\mathbf{b} \in \mathbb{R}^m$
3. Una matriz $m \times n$, $\mathbf{A}$

Con estos elementos, el problema lineal asociado y en forma estándar tiene la siguiente forma. Minimizar

$$Z = \mathbf{c}^T\mathbf{x} \tag{4.8}$$

sujeto a

$$\begin{array}{rcl} \mathbf{Ax} & = & \mathbf{b} \\ \mathbf{x} & \geq & \mathbf{0} \end{array} \tag{4.9}$$

donde $\mathbf{c}^T\mathbf{x}$ indica producto escalar de los vectores $\mathbf{c}$ y $\mathbf{x}$, $\mathbf{Ax}$ es el producto de la matriz $\mathbf{A}$ y el vector $\mathbf{x}$, y $\mathbf{x} \geq \mathbf{0}$ hace que todas la componentes de los vectores factibles sean no negativas. Los problemas de programación lineal se estudian normalmente en esta forma. Típicamente, n es mucho mayor que m. En resumen, un problema de programación lineal se dice que está en forma estándar si y sólo si

1. Es de minimización.
2. Sólo incluye restricciones de igualdad.
3. El vector $\mathbf{b}$ es no negativo.
4. Las variables $\mathbf{x}$ son no negativas.

Antes de analizar el PPL en forma estándar, es conveniente mostrar que cualquier problema expresado en la forma (4.1)–(4.2) también puede expresarse en forma estándar.

4.3.1 Transformación a la forma estándar

Cualquier problema de programación lineal puede expresarse siempre en forma estándar sin más que llevar a cabo una serie de manipulaciones algebraicas:

1. Las variables no restringidas en signo se pueden expresar como diferencias de variables que sí están restringidas en signo, es decir variables no negativas. Si algunas (o todas) de las variables no están restringidas en signo, éstas se pueden expresar mediante sus partes positiva y negativa. Las partes positiva y negativa de la variable x_i, se definen como $x_i^+ = \max\{0, x_i\}$ y $x_i^- = \max\{0, -x_i\}$, respectivamente. Se puede comprobar fácilmente que $x = x_i^+ - x_i^-$, $|x| = x_i^+ + x_i^-$ y que ambas x_i^+ y x_i^- son no negativas. Si el número de variables no restringidas en signo es r, el empleo de la regla anterior supone la necesidad de utilizar r variables adicionales. En el Capitulo 13, sección 13.2.1, se propone una alternativa *mejor*.

2. Las restricciones de desigualdad pueden convertirse en restricciones equivalentes de igualdad introduciendo nuevas variables que se denominan *variables de holgura*:

   - Si
     $$a_{i1}x_1 + a_{i2}x_2 + \cdots + a_{in}x_n \leq b_i$$
     entonces, existe una variable $x_{n+1} \geq 0$ tal que
     $$a_{i1}x_1 + a_{i2}x_2 + \cdots + a_{in}x_n + x_{n+1} = b_i$$

   - Si
     $$a_{i1}x_1 + a_{i2}x_2 + \cdots + a_{in}x_n \geq b_i$$
     entonces, existe una variable $x_{n+1} \geq 0$ tal que
     $$a_{i1}x_1 + a_{i2}x_2 + \cdots + a_{in}x_n - x_{n+1} = b_i$$

3. Un problema de maximización es equivalente a uno de minimización sin más que cambiar el signo de la función objetivo. En particular, maximizar
   $$Z_{max} = \mathbf{c}^T\mathbf{x}$$
   es equivalente a minimizar
   $$Z_{min} = -\mathbf{c}^T\mathbf{x}$$
   si ambos problemas han de cumplir las mismas restricciones. Obsérvese que ambos problemas alcanzan el óptimo en los mismos puntos, pero $Z_{max} = -Z_{min}$.

4. Una restricción con término independiente b no positivo puede reemplazarse por otra equivalente cuyo término independiente es no negativo.

Se describen a continuación algunos ejemplos para ilustrar la transformación a la forma estándar.

Ejemplo 4.5 (forma estándar). La forma estándar del problema de programación lineal: maximizar

$$Z = 2x_1 - 3x_2 + 5x_3$$

sujeto a

$$\begin{array}{rrrcl} x_1 & +x_2 & & \leq & 2 \\ 3x_1 & +x_2 & -x_3 & \geq & 3 \\ & & x_1,\, x_2 & \geq & 0 \end{array} \tag{4.10}$$

es: minimizar

$$Z = -2x_1 + 3x_2 - 5(x_6 - x_7)$$

sujeto a

$$\begin{array}{rrrrrcl} x_1 & +x_2 & & +x_4 & & = & 2 \\ 3x_1 & +x_2 & -(x_6 - x_7) & & -x_5 & = & 3 \\ & & & x_1,\, x_2,\, x_4,\, x_5,\, x_6,\, x_7 & & \geq & 0 \end{array} \tag{4.11}$$

■

Ejemplo 4.6 (variables no negativas: forma estándar). Considérese el siguiente PPL, en el que se maximiza

$$Z = 3x_1 - x_3$$

sujeto a

$$\begin{array}{rrrcr} x_1 & +x_2 & +x_3 & = & 1 \\ x_1 & -x_2 & -x_3 & \leq & 1 \\ x_1 & & +x_3 & \geq & -1 \\ x_1 & & & \geq & 0 \end{array} \tag{4.12}$$

En primer lugar, se cambia el signo de la tercera restricción, en segundo lugar se ha de abordar la no negatividad de las variables. Con este fin, se escribe

$$\begin{array}{rcl} x_2 & = & y_2 - z_2 \\ x_3 & = & y_3 - z_3 \\ y_2, y_3, z_2, z_3 & \geq & 0 \end{array} \tag{4.13}$$

donde $y_2 = x_2^+$, $y_3 = x_3^+$, y $z_2 = x_2^-$, $z_3 = x_3^-$. Por tanto, el problema inicial se transforma en maximizar

$$Z = 3x_1 - y_3 + z_3$$

sujeto a

$$\begin{array}{rrrrrcl} x_1 & +y_2 & -z_2 & +y_3 & -z_3 & = & 1 \\ x_1 & -y_2 & +z_2 & -y_3 & +z_3 & \leq & 1 \\ -x_1 & & & -y_3 & +z_3 & \leq & 1 \\ x_1, & y_2, & z_2, & y_3, & z_3 & \geq & 0 \end{array} \tag{4.14}$$

En segundo lugar, mediante las variables de holgura, u_1 y u_2, las desigualdades se transforman en igualdades

$$\begin{array}{rrrrrrrcl} x_1 & -y_2 & +z_2 & -y_3 & +z_3 & +u_1 & & = & 1 \\ x_1 & & & +y_3 & -z_3 & & -u_2 & = & -1 \end{array} \tag{4.15}$$

Finalmente, se transforma la maximización en minimización cambiando el signo de la función objetivo, por tanto se minimiza

$$Z = -3x_1 + y_3 - z_3$$

sujeto a

$$\begin{array}{rrrrrrrcl} x_1 & +y_2 & -z_2 & +y_3 & -z_3 & & & = & 1 \\ x_1 & -y_2 & +z_2 & -y_3 & +z_3 & +u_1 & & = & 1 \\ -x_1 & & & -y_3 & +z_3 & & +u_2 & = & 1 \\ x_1, & y_2, & z_2, & y_3, & z_3 & u_1 & u_2 & \geq & 0 \end{array} \tag{4.16}$$

Una vez que se ha resuelto este problema, la relación entre las variables x_1, x_2, x_3 y y_2, y_3, z_2, z_3 permite obtener la solución óptima del problema original. Esto es, el vector $(x_1, y_2 - z_2, y_3 - z_3)$ es óptimo para el problema inicial si $(x_1, y_2, y_3, z_2, z_3, u_1, u_2)$ es óptimo para el problema en forma estándar. Obsérvese que las variables de holgura no intervienen en la solución del problema original ya que son variables auxiliares. Finalmente, el valor óptimo de la función objetivo del problema original es el valor óptimo del problema en forma estándar pero cambiado de signo. ∎

4.4 Soluciones básicas

Considérese un problema de programación lineal en forma estándar matricial (4.8)–(4.9), donde se supondrá, sin pérdida de generalidad, que el rango de la matriz $\mathbf{A}$ de dimensión $m \times n$ es m (recuérdese que $m \leq n$), y que el sistema lineal $\mathbf{Ax} = \mathbf{b}$ tiene solución. En cualquier otro caso, el problema lineal es equivalente a otro con menos restricciones, o no tiene solución factible, respectivamente.

Definición 4.4 (matriz básica). *Una submatriz no singular* $\mathbf{B}$ *de dimensión* $m \times m$ *de* $\mathbf{A}$ *se denomina matriz básica o base.* $\mathbf{B}$ *también se denomina matriz básica factible si y solo si* $\mathbf{B}^{-1}\mathbf{b} \geq 0$. ∎

Cada matriz básica tiene un vector asociado que se denomina *solución básica*. El procedimiento para calcular esta solución es el siguiente. Sea $\mathbf{x}_B$ el vector de las variables asociadas a las columnas de $\mathbf{A}$ necesarias para construir $\mathbf{B}$. Las

variables $\mathbf{x}_B$ se denominan *variables básicas* y el resto se denominan *variables no básicas*. Asignando el valor cero a las variables no básicas

$$(\mathbf{B} \quad \mathbf{N}) \begin{pmatrix} \mathbf{x}_B \\ \mathbf{0} \end{pmatrix} = \mathbf{b}, \;\; \mathbf{B}\mathbf{x}_B = \mathbf{b}, \;\; \mathbf{x}_B = \mathbf{B}^{-1}\mathbf{b} \tag{4.17}$$

donde $\mathbf{N}$ es tal que $\mathbf{A} = (\mathbf{B} \quad \mathbf{N})$. Por tanto $\mathbf{B}^{-1}\mathbf{b}$ nos permite obtener la solución básica asociada a $\mathbf{B}$. Si $\mathbf{B}$ es una matriz básica factible, su solución básica se dice que es factible. El número de soluciones básicas factibles de un problema de programación lineal acotado con un número finito de restricciones es siempre finito, y cada una se corresponde con un punto extremo de la región de factibilidad. En concreto, el teorema siguiente establece la relación entre soluciones básicas factibles (SBFs) y puntos extremos.

Teorema 4.1 (caracterización de puntos extremos). *Sea* $S = \{\mathbf{x} : \mathbf{A}\mathbf{x} = \mathbf{b}, \mathbf{x} \geq \mathbf{0}\}$*, donde* $\mathbf{A}$ *es una matrix* $m \times n$ *de rango* m*, y* $\mathbf{b}$ *es un vector de dimensión* m*. Un punto* $\mathbf{x}$ *es punto extremo de* S *si y sólo si* $\mathbf{A}$ *puede descomponerse en* $(\mathbf{B}, \mathbf{N})$ *tal que*

$$\mathbf{x} = \begin{pmatrix} \mathbf{x}_B \\ \mathbf{x}_N \end{pmatrix} = \begin{pmatrix} \mathbf{B}^{-1}\mathbf{b} \\ \mathbf{0} \end{pmatrix}$$

donde $\mathbf{B}$ *es una matriz de dimensión* $m \times m$ *invertible que satisface* $\mathbf{B}^{-1}\mathbf{b} \geq \mathbf{0}$. ∎

Una demostración de este teorema puede encontrarse en el texto de Bazaraa et al. [9]. El teorema siguiente muestra por qué las soluciones básicas factibles son importantes.

Teorema 4.2 (propiedad fundamental de la programación lineal). *Si un problema de programación lineal tiene una solución óptima, es además una solución básica factible.* ∎

Demostración. Una región poliédrica siempre puede escribirse como

$$\mathbf{x} = \sum_i \rho_i \mathbf{v}_i + \sum_j \pi_j \mathbf{w}_j + \sum_k \lambda_k \mathbf{q}_k$$

donde $\rho_i \in \mathbb{R}$, $\pi_j \in \mathbb{R}^+$, y $0 \leq \lambda_k \leq 1; \sum\limits_k \lambda_k = 1$. La región factible de un problema de programación lineal en forma estándar no contiene un espacio vectorial debido a las restricciones $\mathbf{x} \geq \mathbf{0}$. En este caso especial, el conjunto poliédrico puede escribirse como suma de un politopo y un cono; siendo el conjunto mínimo de generadores del politopo el conjunto de puntos extremos, y el mínimo conjunto de generadores del cono el conjunto de direcciones extremas. Por tanto, el valor de la función objetivo a minimizar $Z = \mathbf{c}^T\mathbf{x}$ se puede expresar como

$$Z = \mathbf{c}^T\mathbf{x} = \sum_j \pi_j \mathbf{c}^T \mathbf{w}_j + \sum_k \lambda_k \mathbf{c}^T \mathbf{q}_k$$

Para que este problema tenga solución acotada, ha de cumplirse que

$$\mathbf{c}^T \mathbf{w}_j \geq 0, \ \forall j$$

Si no es así, el valor de la función objetivo puede disminuirse tanto como se quiera sin más que seleccionar valores adecuados para $\pi_j = 0$ y/o λ_k. Por tanto, el valor de la función objetivo pasa a ser

$$Z = \mathbf{c}^T \mathbf{x} = \sum_k \lambda_k \mathbf{c}^T \mathbf{q}_k$$

que alcanza el mínimo para

$$\lambda_s = 1 \text{ para algún } s \text{ tal que } \mathbf{c}^T \mathbf{q}_s = \min(\mathbf{c}^T \mathbf{q}_1, \ldots, \mathbf{c}^T \mathbf{q}_r)$$

donde r es el número de puntos extremos. Lo anterior implica que el mínimo se alcanza en uno de los puntos extremos $\{\mathbf{q}_1, \ldots, \mathbf{q}_r\}$ y, empleando el Teorema 4.1, el punto es una solución básica factible. ∎

4.5 Sensibilidades

A partir de la solución de un PPL se puede extraer información muy relevante sobre sensibilidades. Esto se pone de manifiesto en lo que sigue. Sea B^* la base óptima; entonces

$$\begin{array}{rcl} \mathbf{x}_B^* & = & (\mathbf{B}^*)^{-1}\mathbf{b} \\ z^* & = & \mathbf{c}_B^T \mathbf{x}_B^* \end{array} \tag{4.18}$$

Considérese un cambio marginal (es decir, un cambio que no modifica la base) en el vector de términos independientes $\mathbf{b}$:

$$\mathbf{b}^* \to \mathbf{b}^* + \Delta\mathbf{b}$$

Este cambio en el vector de términos independientes da lugar a cambios en el minimizador y en el valor óptimo de la función objetivo:

$$\begin{array}{c} \mathbf{x}_B^* \to \mathbf{x}_B^* + \Delta\mathbf{x}_B \\ z^* \to z^* + \Delta z \end{array}$$

Puesto que las ecuaciones (4.18) son lineales, se puede escribir

$$\begin{array}{c} \Delta\mathbf{x}_B = (\mathbf{B}^*)^{-1}\Delta\mathbf{b} \\ \Delta z = \mathbf{c}_B^T \Delta\mathbf{x}_B \end{array}$$

Combinando las ecuaciones anteriores, se obtiene

$$\Delta z = \mathbf{c}_B^T \Delta\mathbf{x}_B = \mathbf{c}_B^T (\mathbf{B}^*)^{-1}\Delta\mathbf{b}$$

Definiendo

$$\boldsymbol{\lambda}^{*T} = \mathbf{c}_B^T (\mathbf{B}^*)^{-1}$$

la ecuación anterior pasa a

$$\Delta z = \boldsymbol{\lambda}^{*T} \Delta \mathbf{b}$$

y para una coordenada arbitraria j la expresión anterior tiene la forma

$$\lambda_j^* = \frac{\Delta z}{\Delta b_j}$$

que indica que λ_j^* proporciona el cambio en el valor óptimo de la función objetivo como resultado de un cambio marginal en la componente j del vector de términos independientes $\mathbf{b}$. Estos parámetros de sensibilidad juegan un papel fundamental en aplicaciones de ingeniería y científicas. Como se verá en las secciones siguientes, los parámetros de sensibilidad son de hecho variables duales.

4.6 Dualidad

Esta sección trata la dualidad en programación lineal, su concepto y significado. Tras formular el problema dual de un problema de programación lineal, se establece la relación matemática entre ambos. Se emplean diversos ejemplos para ilustrar el importante concepto de la dualidad. Dado un problema de programación lineal, denominado *problema primal*, existe otro problema de programación lineal, denominado *problema dual*, íntimamente relacionado con él. Se dice que ambos problemas son mutuamente duales. Bajo ciertas hipótesis, los problemas primal y dual dan lugar al mismo valor óptimo de la función objetivo, y por tanto se puede resolver indirectamente el problema primal resolviendo el problema dual. Esto puede suponer una ventaja computacional relevante.

Definición 4.5 (problema dual). *Dado el problema de programación lineal minimizar*

$$Z = \mathbf{c}^T \mathbf{x}$$

sujeto a

$$\begin{array}{rcl} \mathbf{A}\mathbf{x} & \geq & \mathbf{b} \\ \mathbf{x} & \geq & \mathbf{0} \end{array} \tag{4.19}$$

su problema dual es maximizar

$$Z = \mathbf{b}^T \mathbf{y}$$

sujeto a

$$\begin{array}{rcl} \mathbf{A}^T\mathbf{y} & \leq & \mathbf{c} \\ \mathbf{y} & \geq & \mathbf{0} \end{array} \tag{4.20}$$

donde $\mathbf{y} = (y_1, \ldots, y_m)^T$ *se denominan variables duales.* ∎

Se denomina al primer problema problema primal, y al segundo, su dual. Obsérvese que los mismos elementos (la matriz $\mathbf{A}$, y los vectores $\mathbf{b}$ y $\mathbf{c}$) configuran ambos problemas. El problema primal no se ha escrito en forma estándar, sino en una forma que nos permite apreciar la simetría entre ambos problemas, y mostrar así que el dual del dual es el primal.

Nota 4.1 *La dualidad es una relación simétrica, esto es, si el problema D es el dual del problema P, entonces P es el dual de D.* ■

Para comprobar lo anterior, se escribe el problema dual anterior como un problema de minimización con restricciones de la forma $\geq$. Minimizar

$$Z = -\mathbf{b}^T\mathbf{y}$$

sujeto a

$$\begin{array}{rcl} -\mathbf{A}^T\mathbf{y} & \geq & -\mathbf{c} \\ \mathbf{y} & \geq & \mathbf{0} \end{array} \tag{4.21}$$

Entonces, su dual es maximizar

$$Z = -\mathbf{c}^T\mathbf{x}$$

sujeto a

$$\begin{array}{rcl} -\mathbf{A}\mathbf{x} & \leq & -\mathbf{b} \\ \mathbf{x} & \geq & \mathbf{0} \end{array} \tag{4.22}$$

que es equivalente al problema primal original.

Nota 4.2 *Como puede observarse, cada restricción del problema primal tiene asociada una variable del problema dual; los coeficientes de la función objetivo del problema primal son los términos independientes de las restricciones del problema dual y viceversa; y la matriz de restricciones del problema dual es la traspuesta de la matriz de restricciones del problema primal. Además, el problema primal es de minimización y el dual de maximización.* ■

4.6.1 Obtención del dual a partir del primal en forma estándar

A continuación, se obtiene el problema dual a partir del problema primal en forma estándar. Para hacer esto basta con aplicar las relaciones primal-dual de la sección 4.6. Considérese el PPL minimizar

$$Z = \mathbf{c}^T\mathbf{x}$$

sujeto a

$$\begin{array}{rcl} \mathbf{A}\mathbf{x} & = & \mathbf{b} \\ \mathbf{x} & \geq & \mathbf{0} \end{array}$$

La igualdad $\mathbf{A}\mathbf{x} = \mathbf{b}$ puede reemplazarse por las desigualdades $\mathbf{A}\mathbf{x} \geq \mathbf{b}$ y $-\mathbf{A}\mathbf{x} \geq -\mathbf{b}$. Entonces, puede escribirse el problema como: minimizar

$$Z = \mathbf{c}^T\mathbf{x}$$

sujeto a

$$\begin{array}{rcl} \begin{pmatrix} \mathbf{A} \\ -\mathbf{A} \end{pmatrix}\mathbf{x} & \geq & \begin{pmatrix} \mathbf{b} \\ -\mathbf{b} \end{pmatrix} \\ \mathbf{x} & \geq & \mathbf{0} \end{array}$$

El dual de este problema es maximizar

$$Z = \mathbf{b}^T\mathbf{y}^{(1)} - \mathbf{b}^T\mathbf{y}^{(2)} = \mathbf{b}^T\mathbf{y}$$

donde $\mathbf{y} = \mathbf{y}^{(1)} - \mathbf{y}^{(2)}$ no está restringida en signo, sujeto a

$$\begin{pmatrix} \mathbf{A}^T & -\mathbf{A}^T \end{pmatrix} \begin{pmatrix} \mathbf{y}^{(1)} \\ \mathbf{y}^{(2)} \end{pmatrix} = \mathbf{A}^T\mathbf{y} \le \mathbf{c}$$

Esta es la forma del problema dual cuando el problema primal se expresa en forma estándar.

4.6.2 Obtención del problema dual

Un problema de programación lineal de la forma (4.19) tiene asociado un problema dual que puede formularse según las reglas siguientes:

Regla 1. Una restricción de igualdad en el primal (dual) hace que la correspondiente variable dual (primal) no esté restringida en signo.

Regla 2. Una restricción de desigualdad $\geq$ ($\leq$) en el primal (dual) da lugar a una variable dual (primal) no negativa.

Regla 3. Una restricción de desigualdad $\leq$ ($\geq$) en el primal (dual) da lugar a una variable dual (primal) no positiva.

Regla 4. Una variable no negativa primal (dual) da lugar a una restricción de desigualdad $\leq$ ($\geq$) en el problema dual (primal).

Regla 5. Una variable primal (dual) no positiva da lugar a una restricción de desigualdad $\geq$ ($\leq$) en el problema dual (primal).

Regla 6. Una variable no restringida en signo del problema primal (dual) da lugar a una restricción de igualdad en el dual (primal).

Ejemplo 4.7 (problema dual). El dual del problema de programación lineal minimizar

$$Z = x_1 + x_2 - x_3$$

sujeto a

$$\begin{array}{rrrcl} 2x_1 & +x_2 & & \geq & 3 \\ x_1 & & -x_3 & = & 2 \\ & & x_3 & \geq & 0 \end{array} \tag{4.23}$$

es maximizar

$$Z = 3y_1 + 2y_2$$

sujeto a

$$
\begin{array}{rrcr}
2y_1 & +y_2 & = & 1 \\
y_1 & & = & 1 \\
 & -y_2 & \leq & -1 \\
 & y_1 & \geq & 0
\end{array}
\tag{4.24}
$$

Para obtenerlo se aplican las reglas anteriores de la forma siguiente:

Regla 1. Puesto que la segunda restricción del problema primal es de igualdad, la segunda variable dual y_2 no está restringida en signo.

Regla 2. Puesto que la primera restricción del problema primal es de desigualdad $\geq$, la primera variable dual y_1 es no negativa.

Regla 3. Puesto que la tercera variable primal x_3 está restringida en signo, la tercera restricción dual es de desigualdad $\leq$.

Regla 4. Puesto que las variables primales primera y segunda x_1 y x_2 no están restringidas en signo, las restricciones duales primera y segunda son de igualdad.

Aplicando las mismas reglas, se puede obtener el problema primal del dual, lo que se muestra a continuación:

Regla 1. Dado que las restricciones primera y segunda del problema dual son de igualdad, las variables primales primera y segunda x_1 y x_2 no están restringidas en signo.

Regla 2. Dado que al tercera restricción del problema dual es de desigualdad $\leq$, la tercera variable primal x_3 es no negativa.

Regla 3. Dado que la primera variable dual y_1 está restringida en signo, la primera restricción primal es de desigualdad $\geq$.

Regla 4. Puesto que la segunda variable dual y_2 no está restringida en signo, la segunda restricción primal es de igualdad.

■

4.6.3 Teoremas de dualidad

La importancia del problema dual se establece en los siguientes teoremas.

Lema 4.1 (lema de dualidad débil) *Sea P un PPL, y D su dual. Sea* $\mathbf{x}$ *una solución factible de P e* $\mathbf{y}$ *una solución factible de D. Entonces*

$$\mathbf{b}^T\mathbf{y} \leq \mathbf{c}^T\mathbf{x}$$

Demostración. La demostración es sencilla. Si $\mathbf{x}$ e $\mathbf{y}$ son factibles respectivamente para P y D, entonces

$$\mathbf{A}\mathbf{x} = \mathbf{b}, \ \ \mathbf{x} \geq \mathbf{0}, \ \ \mathbf{A}^T\mathbf{y} \leq \mathbf{c}$$

Obsérvese que debido a la no negatividad de $\mathbf{x}$,

$$\mathbf{b}^T\mathbf{y} = \mathbf{x}^T\mathbf{A}^T\mathbf{y} \leq \mathbf{x}^T\mathbf{c} = \mathbf{c}^T\mathbf{x}$$

∎

Corolario 4.1 *Si $\mathbf{b}^T\tilde{\mathbf{y}} = \mathbf{c}^T\tilde{\mathbf{x}}$ para dos vectores $\tilde{\mathbf{x}}$ e $\tilde{\mathbf{y}}$, factibles en P y D, respectivamente, entonces*

$$\mathbf{c}^T\tilde{\mathbf{x}} = \mathbf{b}^T\tilde{\mathbf{y}} \leq \max_{\mathbf{y}}\{\mathbf{b}^T\mathbf{y}|\mathbf{A}^T\mathbf{y} \leq \mathbf{c}\} \leq \min_{\mathbf{x}}\{\mathbf{c}^T\mathbf{x}|\mathbf{A}\mathbf{x} = \mathbf{b}, \ \ \mathbf{x} \geq \mathbf{0}\} \leq \mathbf{c}^T\tilde{\mathbf{x}} = \mathbf{b}^T\tilde{\mathbf{y}}$$

Demostración. La demostración es simple si se comprueba que

$$\max_{\mathbf{y}}\{\mathbf{b}^T\mathbf{y}|\mathbf{A}^T\mathbf{y} \leq \mathbf{c}\} \leq \min_{\mathbf{x}}\{\mathbf{c}^T\mathbf{x}|\mathbf{A}\mathbf{x} = \mathbf{b}, \ \ \mathbf{x} \geq \mathbf{0}\}$$

∎

Por tanto, todas las desigualdades son de hecho igualdades y $\tilde{\mathbf{x}}$ e $\tilde{\mathbf{y}}$ deben ser soluciones óptimas de P y D respectivamente, tal como establecía la hipótesis inicial. El teorema de dualidad fuerte establece que los problemas P y D tienen, en general, soluciones óptimas simultáneamente.

Teorema 4.3 (teorema de dualidad). *Si $\tilde{\mathbf{x}}$ es una solución óptima de P, existe una solución óptima $\tilde{\mathbf{y}}$ para D, y el mínimo de P y el máximo de D presentan el mismo valor de la función objetivo $\mathbf{b}^T\tilde{\mathbf{y}} = \mathbf{c}^T\tilde{\mathbf{x}}$. Recíprocamente, si $\tilde{\mathbf{y}}$ es una solución óptima de D, existe una solución óptima de P, $\tilde{\mathbf{x}}$, y nuevamente los valores mínimo y máximo de P y D dan lugar a un valor común de la función objetivo $\mathbf{b}^T\tilde{\mathbf{y}} = \mathbf{c}^T\tilde{\mathbf{x}}$. En otro caso, o un conjunto factible está vacío o lo están los dos.* ∎

En resumen, si P es un PPL y D es su dual, una de las siguientes afirmaciones es cierta:

1. Ambos problemas tienen solución óptima y los valores óptimos de las funciones objetivo respectivas coinciden.

2. Uno de los problemas no está acotado y el otro tiene una región factible vacía.

Ejemplo 4.8 (problemas primal y dual del carpintero). Un carpintero modesto fabrica dos tipos de mesas de madera. Cada mesa del tipo 1 necesita 4 horas de mecanizado primario (preparación de piezas) y 4 horas de mecanizado secundario (ensamblado y barnizado). Análogamente, cada mesa del tipo 2 necesita 3 horas de mecanizado primario y 7 horas de mecanizado secundario.

Tabla 4.1: Datos para el problema del carpintero

	Tipo de mesa		Disponibilidad de horas máquina por día
	1	2	
Mecanizado primario	4	3	40
Mecanizado secundario	4	7	56
Beneficio (dólares)	70	90	

Las disponibilidades diarias de mecanizados primario y secundario son respectivamente de 40 y 56 horas-máquina. La venta de una mesa del tipo 1 reporta un beneficio de 70 dólares, mientras que la venta de una mesa del tipo 2 de 90 dólares. Estos datos se resumen en la tabla 4.1. El objeto de este problema es determinar el número de mesas de cada tipo que han de producirse diariamente para maximizar el beneficio obtenido. Este problema puede formularse como un problema de programación lineal que maximiza

$$Z = 70x_1 + 90x_2$$

sujeto a

$$\begin{array}{rrcr} 4x_1 & +3x_2 & \leq & 40 \\ 4x_1 & +7x_2 & \leq & 56 \\ & x_1, x_2 & \geq & 0 \end{array} \qquad (4.25)$$

donde x_1 y x_2 son las cantidades diarias de mesas a fabricar de los tipos 1 y 2 respectivamente. La solución óptima de este problema, como se observa en la figura 4.4, establece que han de producirse diariamente 7 y 4 sillas de los tipos 1 y 2 respectivamente, lo que da lugar a un beneficio de 850 dólares. Este resultado indica que ambos recursos de mecanizado (primario y secundario) están plenamente utilizados porque las restricciones relacionadas con ellos están ambas activas. Por otra parte, considérese que quiere aumentarse el beneficio diario. Para ello es necesario aumentar la capacidad productiva. Considérese que la capacidad de mecanizado secundario puede aumentarse cada día de 56 a 72 horas de máquina. ¿Cómo afecta esta ampliación de capacidad a los beneficios diarios? La solución puede obtenerse resolviendo el siguiente problema en el que se maximiza

$$Z = 70x_1 + 90x_2$$

sujeto a

$$\begin{array}{rrcr} 4x_1 & +3x_2 & \leq & 40 \\ 4x_1 & +7x_2 & \leq & 72 \\ & x_1, x_2 & \geq & 0 \end{array} \qquad (4.26)$$

En este caso la solución óptima es $x_1 = 4$ y $x_2 = 8$ con un beneficio máximo diario de 1000 dólares. Este solución indica que el beneficio diario crece en

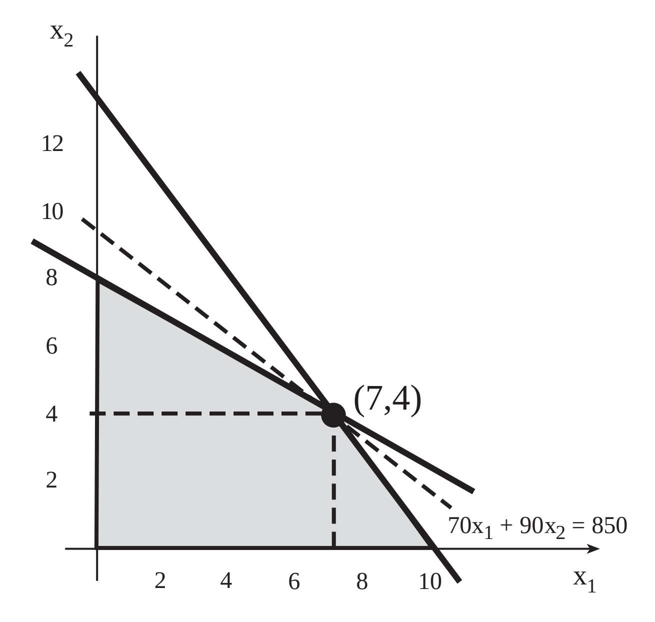


Figura 4.4: Análisis gráfico del problema del carpintero.

150 dólares y la capacidad de mecanizado secundario crece en $72 - 56 = 16$ horas máquina. El ratio 1000-850/16=150/16 = 75/8 dólares, al que la función objetivo crece al crecer la capacidad de mecanizado secundario 1 hora, se denomina *sensibilidad o precio sombra* (también *precio dual*) de la capacidad de mecanizado secundario (véase la sección 4.5). En general el precio sombra de una restricción proporciona el cambio en el valor de la función objetivo como resultado de un cambio unitario en el término independiente de la restricción, suponiendo que el resto de parámetros del problema permanecen inalterados. En muchos problemas de programación lineal los precios sombra son tan importantes como la solución del problema, ya que proporcionan información sobre el efecto en la función objetivo de cambios en los recursos disponibles. Los precios sombra pueden obtenerse resolviendo el *problema dual*.
El problema dual del problema del carpintero (4.25) se formula a continuación.
Minimizar

$$Z = 40y_1 + 56y_2$$

sujeto a

$$\begin{array}{rrcr} 4y_1 & +4y_2 & \geq & 70 \\ 3y_1 & +7y_2 & \geq & 90 \\ & y_1, y_2 & \geq & 0 \end{array} \tag{4.27}$$

La solución óptima de este problema es $y_1 = 65/8$, $y_2 = 75/8$, y el valor óptimo de la función objetivo es 850. Obsérvese que y_1 y y_2 son los precios sombra de las capacidades de mecanizado primario y secundario, respectivamente, y que

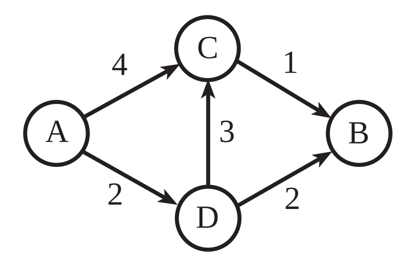


Figura 4.5: Red de comunicaciones del Ejemplo 4.9.

los valores óptimos de la función objetivo de (4.25) y (4.27) coinciden. El problema dual (4.27) puede interpretarse de la siguiente manera. Considérese que el objetivo es vender tiempo de mecanizado primario y secundario y supóngase que de esta forma se obtienen al menos el mismo nivel de beneficios que haciendo mesas. En esta situación vender tiempo de mecanizado y hacer mesas han de ser actividades igualmente lucrativas. Las variables y_1 y y_2 variables representan los precios de venta de una hora de mecanizados primario y secundario respectivamente. Para preservar la competitividad del negocio, el beneficio diario ha de minimizarse, esto es minimizar la función $40y_1 + 56y_2$, donde 40 y 56 representan respectivamente la disponibilidad diaria en horas de mecanizado primario y secundario respectivamente. Las restricciones (4.27) establecen que el coste de las horas de mecanizado primario y secundario para producir una mesa de cada tipo no debe superar el beneficio que se obtiene por venta de la misma; y que los precios son cantidades no negativas. ■

Ejemplo 4.9 (red de comunicaciones). El objeto de este problema es enviar un mensaje a través de una red de comunicaciones desde el nudo A al nudo B incurriendo en un coste mínimo. La red está compuesta por cuatro nudos A, B, C y D, y hay líneas de conexión entre A y C, A y D, D y C, C y B, y D y B, con coste unitarios por envío de mensajes de 4, 2, 3, 1, y 2, respectivamente (véase la figura 4.5).

Primal

Un problema primal que describe la situación previa consiste en minimizar

$$Z = 4x_{AC} + 2x_{AD} + 3x_{DC} + x_{CB} + 2x_{DB}$$

donde cada variable x_{PQ} denota la fracción de mensaje a través de la red que va por el canal de comunicación que une los nudos P y Q. La formulación anterior permite minimizar el coste total de enviar un mensaje. Han de cumplirse las siguientes restricciones:

1. Ninguna fracción de mensaje puede perderse en el nudo C:

$$x_{AC} + x_{DC} \quad = \quad x_{CB} \tag{4.28}$$

2. Ninguna fracción de mensaje puede perderse en el nudo D:

$$x_{AD} = x_{DC} + x_{DB} \tag{4.29}$$

3. El mensaje debe llegar íntegro a B:

$$1 = x_{CB} + x_{DB} \tag{4.30}$$

Este conjunto de restricciones puede escribirse en forma matricial

$$\begin{pmatrix} 1 & 0 & 1 & -1 & 0 \\ 0 & 1 & -1 & 0 & -1 \\ 0 & 0 & 0 & 1 & 1 \end{pmatrix} \begin{pmatrix} x_{AC} \\ x_{AD} \\ x_{DC} \\ x_{CB} \\ x_{DB} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \tag{4.31}$$

Obsérvese que si se suman las dos últimas restricciones y al resultado se le resta la primera, se obtiene $x_{AC} + x_{AD} = 1$. Esto indica que el mensaje íntegro parte de A. Puesto que esta última restricción se obtiene como combinación lineal de las anteriores, no se incluye para evitar redundancias.

Dual

Alternativamente, puede pensarse en la posibilidad de comprar fracciones de la información de los mensajes en determinados nudos y venderla en otros. Una aproximación dual del problema anterior, consiste pues en determinar los precios de compraventa y_A, y_B, y_C, y_D de las fracciones de información de los mensajes en los nudos. Se considera que las fracciones de información se compran en unos nudos y se venden en otros. El beneficio neto de una transacción viene dado por la diferencia de precios entre los nudos de entrada y salida de la fracción de información. Si se normaliza suponiendo $y_A = 0$, los beneficios son

$$\begin{array}{l} y_C - y_A = y_C \\ y_D - y_A = y_D \\ y_C - y_D \\ y_B - y_C \\ y_B - y_D \end{array}$$

Debe imponerse que estos beneficios no superen los precios de transmisión de mensajes ya conocidos:

$$\begin{array}{rcl} y_C & \leq & 4 \\ y_D & \leq & 2 \\ y_C - y_D & \leq & 3 \\ y_B - y_C & \leq & 1 \\ y_B - y_D & \leq & 2 \end{array}$$

Estas restricciones pueden escribirse como

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & -1 & 0 \\ -1 & 0 & 1 \\ 0 & -1 & 1 \end{pmatrix} \begin{pmatrix} y_C \\ y_D \\ y_B \end{pmatrix} \leq \begin{pmatrix} 4 \\ 2 \\ 3 \\ 1 \\ 2 \end{pmatrix} \tag{4.32}$$

Se trata pues de maximizar los ingresos obtenidos cuando un mensaje va del nudo A al nudo B, esto es, maximizar y_B (de hecho, $y_B - y_A$), sujeto a las restricciones anteriores. Las formulaciones anteriores son respectivamente la primal y la dual. ■

Ejercicios

4.1 Determínese la región factible y los puntos extremos del siguiente problema de programación lineal. Maximizar

$$Z = -4x_1 + 7x_2$$

sujeto a

$$\begin{array}{rrcl} x_1 & +x_2 & \geq & 3 \\ -x_1 & +x_2 & \leq & 3 \\ 2x_1 & +x_2 & \leq & 8 \\ & x_1, x_2 & \geq & 0 \end{array}$$

Obténgase la solución gráficamente.

4.2 Obténganse las soluciones de los siguientes problemas gráficamente.

(a) Maximizar

$$Z = 2x_1 + 6x_2$$

sujeto a

$$\begin{array}{rcl} -x_1 + x_2 & \leq & 1 \\ 2x_1 + x_2 & \leq & 2 \\ x_1 & \geq & 0 \\ x_2 & \geq & 0 \end{array}$$

(b) Minimizar

$$Z = -3x_1 + 2x_2$$

sujeto a

$$\begin{array}{c} x_1 + x_2 \leq 5 \\ 0 \leq x_1 \leq 4 \\ 1 \leq x_2 \leq 6 \end{array}$$

4.3 Resuélvase, gráficamente, el siguiente PPL. Minimizar

$$Z = 2x_1 + x_2$$

sujeto a

$$\begin{array}{rrcr} 3x_1 & +x_2 & \geq & 3 \\ 4x_1 & +3x_2 & \geq & 6 \\ x_1 & +2x_2 & \geq & 2 \\ & x_1,\, x_2 & \geq & 0 \end{array}$$

4.4 Formúlese y resuélvase el problema dual del problema de programación lineal siguiente. Maximizar

$$Z = 3x_1 - 4x_2 + 9x_3 + x_4$$

sujeto a

$$\begin{array}{rrrcr} x_1 & -5x_2 & +x_3 & \geq & 0 \\ 3x_1 & -5x_2 & +x_3 & \geq & 10 \\ & & x_2,\, x_3 & \geq & 0 \\ & & x_4 & \leq & 0 \end{array}$$

4.5 Considérese el siguiente problema. Maximizar

$$Z = 8x_1 - 4x_2$$

sujeto a

$$\begin{array}{rrcr} 4x_1 & -5x_2 & \leq & 2 \\ 2x_1 & +4x_2 & \geq & 6 \\ x_1 & -6x_2 & \leq & -14 \end{array}$$

(a) Dibújese la región factible e identifíquense sus puntos extremos.

(b) Determínese un punto de la región factible en el que la función objetivo alcanza un máximo.

4.6 Demuéstrese que el problema siguiente no tiene un valor óptimo finito de la función objetivo. Minimizar

$$Z = -6x_1 + 2x_2$$

sujeto a

$$\begin{array}{rrcr} -3x_1 & +x_2 & \leq & 6 \\ 3x_1 & +5x_2 & \geq & 15 \\ & x_1,\, x_2 & \geq & 0 \end{array}$$

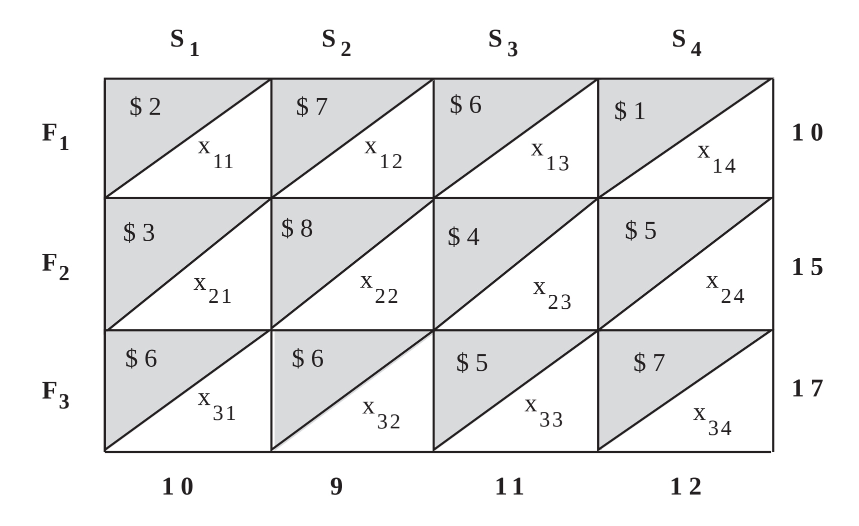


Figura 4.6: Problema de transporte en el que se muestran orígenes y destinos.

Tabla 4.2: Problema de distribución

	Centros de consumo		
Plantas de producción	Centro_1	Centro_2	Centro_3
Planta_1	1	3	5
Planta_2	2	5	4

4.7 Un productor de frigoríficos dispone de tres fábricas que suministran cuatro cadenas de distribución. Considérese el problema de transporte que se muestra en la figura 4.6. Esto es, interesa suministrar 10 unidades desde F_1, 15 desde F_2, y 17 desde F_3, y recibir 10 unidades en S_1, 9 en S_2, 11 en S_3, y 12 en S_4. Sea x_{ij} el número de unidades a transportar desde la fábrica i a la cadena de distribución j. Formúlese el problema de programación lineal asociado.

4.8 Una compañía que produce un producto P dispone de dos plantas de producción. Cada planta produce 90 toneladas del producto P al mes, que se distribuyen en tres centros de consumo. La tabla 4.2 muestra el coste de transportar 1 tonelada del producto P desde una determinada planta de producción a un centro de consumo. El objetivo de la empresa es enviar la misma cantidad de producto a cada centro de consumo maximizando sus beneficios. Formúlese un problema de programación lineal al efecto.

4.9 Una empresa produce 120 unidades de producto A y 360 unidades de producto B cada día. Estos productos han de someterse a control de calidad, siendo la capacidad de control de 200 unidades al día. El producto A se vende en el mercado a un precio 4 veces superior al precio del producto

B. Determínese la producción de la empresa que hace posible maximizar el beneficio.

4.10 Considérese el problema: maximizar

$$Z = \frac{\mathbf{p}^T\mathbf{x} + \alpha}{\mathbf{q}^t\mathbf{x} + \beta}$$

sujeto a

$$\begin{aligned} \mathbf{A}\mathbf{x} &= \mathbf{b} \\ \mathbf{x} &\geq \mathbf{0} \end{aligned}$$

donde $\mathbf{p}$ y $\mathbf{q}$ son n vectores, $\mathbf{b}$ es un m vector, $\mathbf{A}$ es una matriz de dimensión $m \times n$, y α y β son escalares. Supóngase que la región de factibilidad de este problema está acotada y supóngase que $\mathbf{q}^T\mathbf{x} + \beta > 0$ para cada punto factible $\mathbf{x}$. Este problema se denomina *problema de programación lineal fraccional* (PPLF).

(a) Demuéstrese que la transformación

$$z = \frac{1}{\mathbf{q}^T\mathbf{x} + \beta}; \;\; \mathbf{y} = z\mathbf{x}$$

da lugar a un problema equivalente de programación lineal.

(b) Supóngase que el valor óptimo de la función objetivo del PPLF es Z^*. Demuéstrese que el siguiente problema de programación lineal tiene las mismas soluciones. Minimizar

$$Z = \mathbf{p}^T\mathbf{x} + \alpha - Z^*(\mathbf{q}^t\mathbf{x} + \beta)$$

sujeto a

$$\begin{aligned} \mathbf{A}\mathbf{x} &= \mathbf{b} \\ \mathbf{x} &\geq \mathbf{0} \end{aligned}$$

Capítulo 5

El conjunto de soluciones factibles

5.1 Introducción y motivación

Como se estableció en el capítulo 1, un problema de programación matemática consta de los siguientes elementos: datos, variables, restricciones y función a optimizar. En este capítulo se estudia el conjunto de restricciones.

Un PPL se dice que está *bien formulado* si tiene una solución acotada. Si no tiene solución es porque está restringido en exceso. Por tanto, para que el problema esté bien formulado es crucial que las restricciones se elijan adecuadamente.

Las restricciones son el conjunto de condiciones que toda solución candidata a óptima debe cumplir. Estas condiciones están asociadas a la realidad física, económica o de ingeniería en la que surge el problema. Este conjunto de restricciones define el *conjunto factible*, esto es, el conjunto de soluciones que satisfacen las restricciones. Este conjunto tiene interés en sí mismo ya que engloba todas las soluciones que son de interés real y entre las que están las soluciones óptimas.

En los capítulos previos la atención se ha centrado en determinar la solución óptima de una función objetivo cumpliendo un conjunto de restricciones. Esto implica seleccionar de entre todos los vectores que cumplen las restricciones, aquél que maximiza la función objetivo. Por tanto, los métodos previamente estudiados sólo permiten determinar una única solución del problema, según el criterio (de maximización o minimización) que se establece a partir de la función objetivo. Puesto que la función objetivo es independiente del conjunto de restricciones, en este capítulo se trata exclusivamente este conjunto de restricciones.

Un conocimiento detallado de los problemas de programación matemática requiere un conocimiento preciso de las estructuras de las posibles regiones de factibilidad. Las restricciones a que da lugar un conjunto de igualdades y desigualdades lineales originan soluciones factibles que tienen la estructura de es-

Tabla 5.1: Estructuras que surgen en la resolución de sistemas lineales de ecuaciones y desigualdades, y su representación

Estructura algebraica	Definición en función de las restricciones	Representación interna
Espacio vectorial	$\mathbf{Hx} = \mathbf{0}$	$\mathbf{x} = \sum_i \rho_i \mathbf{v}_i;\ \rho_i \in \mathbb{R}$
Espacio afín	$\mathbf{Hx} = \mathbf{a}$	$\mathbf{x} = \mathbf{q} + \sum_i \rho_i \mathbf{v}_i;\ \rho_i \in \mathbb{R}$
Cono	$\mathbf{Hx} \leq \mathbf{0}$	$\mathbf{x} = \sum_j \pi_j \mathbf{w}_j;\ \pi_j \geq 0$
Politopo	$\mathbf{Hx} \leq \mathbf{a}$	$\mathbf{x} = \sum_k \lambda_k \mathbf{q}_k;\ \lambda_k \geq 0;$ $\sum_k \lambda_k = 1$
Poliedro	$\mathbf{Hx} \leq \mathbf{a}$	$\mathbf{x} = \sum_i \rho_i \mathbf{v}_i + \sum_j \pi_j \mathbf{w}_j + \sum_k \lambda_k \mathbf{q}_k;$ $\rho_i \in \mathbb{R};\ \pi_j \geq 0;\ \lambda_k \geq 0;\ \sum_k \lambda_k = 1$

pacios vectoriales, conos, politopos y poliedros (véase la Tabla 5.1). Por tanto, conocer estas estructuras algebraicas es importante para comprender el comportamiento de las soluciones de los problemas en los que intervienen.

Como se muestra en la Tabla 5.1, el conjunto de soluciones factibles puede escribirse de dos maneras: (1) mediante un conjunto de restricciones y (2) mediante una representación interna. La representación mediante restricciones es la forma natural en la que se formula el problema. Sin embargo, partiendo de esta formulación no es fácil encontrar soluciones factibles. De hecho, no es fácil encontrar un vector $\mathbf{x}$ que satisfaga las restricciones. Por el contrario, la representación interna permite generar todas las soluciones factibles sin más que encontrar los valores apropiados de los coeficientes ρ_i, π_j, y λ_k.

Para clarificar lo anterior y motivar el material de este capítulo, se emplea un simple ejemplo que consiste en un cubo. Si, en primer lugar, se considera un problema tridimensional sin restricciones, el conjunto de soluciones factibles es el espacio vectorial

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \rho_1 \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} + \rho_2 \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} + \rho_3 \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

Considérese el conjunto de restricciones

$$\begin{array}{c} 0 \leq x \leq 1 \\ 0 \leq y \leq 1 \\ 0 \leq z \leq 1 \end{array} \tag{5.1}$$

que en conjunto definen una región cúbica (véase la figura 5.1(f)).

Si partiendo de un conjunto vacío, las restricciones (5.1) se incorporan una a una, las soluciones que progresivamente se obtienen se ilustran en la figura 5.1 y en la Tabla 5.2, que muestran los correspondientes sistemas de ecuaciones y sus soluciones. Los coeficientes ρ, π, y λ son números reales no restringidos en signo, no negativos y no negativos, que suman 1, respectivamente. Estos coeficientes permiten escribir combinaciones lineales, combinaciones no negativas y combinaciones convexas de vectores, que dan lugar a espacios vectoriales, conos y politopos, respectivamente. Por ejemplo, al introducir la primera restricción, $z \geq 0$, se fuerza a que todos los valores de z sean no negativos. Por tanto, en vez de emplear un coeficiente ρ_1 (real) se emplea un coeficiente π_1 (no negativo) (véase la figura 5.1(a) y la Tabla 5.2, columna primera). A continuación se introduce la restricción $z \leq 1$, que fuerza a que los valores de π_1 estén entre cero y uno, y se convierte, por tanto, en λ_1, que puede variar entre cero y uno (véase la figura 5.1(b) y la Tabla 5.2, fila segunda). Al introducir la restricción $y \geq 0$, el coeficiente ρ_2 se transforma en el coeficiente π_1 (véase la figura 5.1(c) y la Tabla 5.2, fila tercera). Se sugiere al lector explorar los cambios que tienen lugar cuando se introducen las tres restricciones restantes $x \geq 0$, $y \leq 1$, y $x \leq 1$. El resultado final es la región que se muestra en la figura 5.1(a)–(f) donde se muestra un espacio vectorial y un cono, un espacio vectorial y un politopo, un espacio vectorial, un cono y un politopo, un cono y un politipo, y un politopo, respectivamente.

Aunque obtener el conjunto de soluciones factibles de un cubo es sencillo, obtener en general el conjunto de soluciones factibles asociadas a un conjunto de restricciones lineales (que son las tratadas en este capítulo) no es sencillo. Para resolver este problema, Castillo et al. [21] propusieron el algoritmo Γ, que se muestra en el Apéndice A.

Este capítulo está organizado de la manera siguiente. En la Sección 5.2 se introducen los conjuntos convexos, y los puntos y direcciones extremas; se muestra asimismo cómo un conjunto convexo puede expresarse en función de sus puntos y direcciones extremas. A continuación, se tratan casos especiales de conjuntos convexos, como son los espacios vectoriales, los conos poliédricos convexos, los politopos y los poliedros en las Secciones 5.4–5.7, respectivamente, dado que aparecen en el estudio de las restricciones lineales.

5.2 Conjuntos convexos

En esta sección se estudian los conjuntos convexos. Inicialmente se define el conjunto convexo y se establecen sus propiedades fundamentales; definiéndose

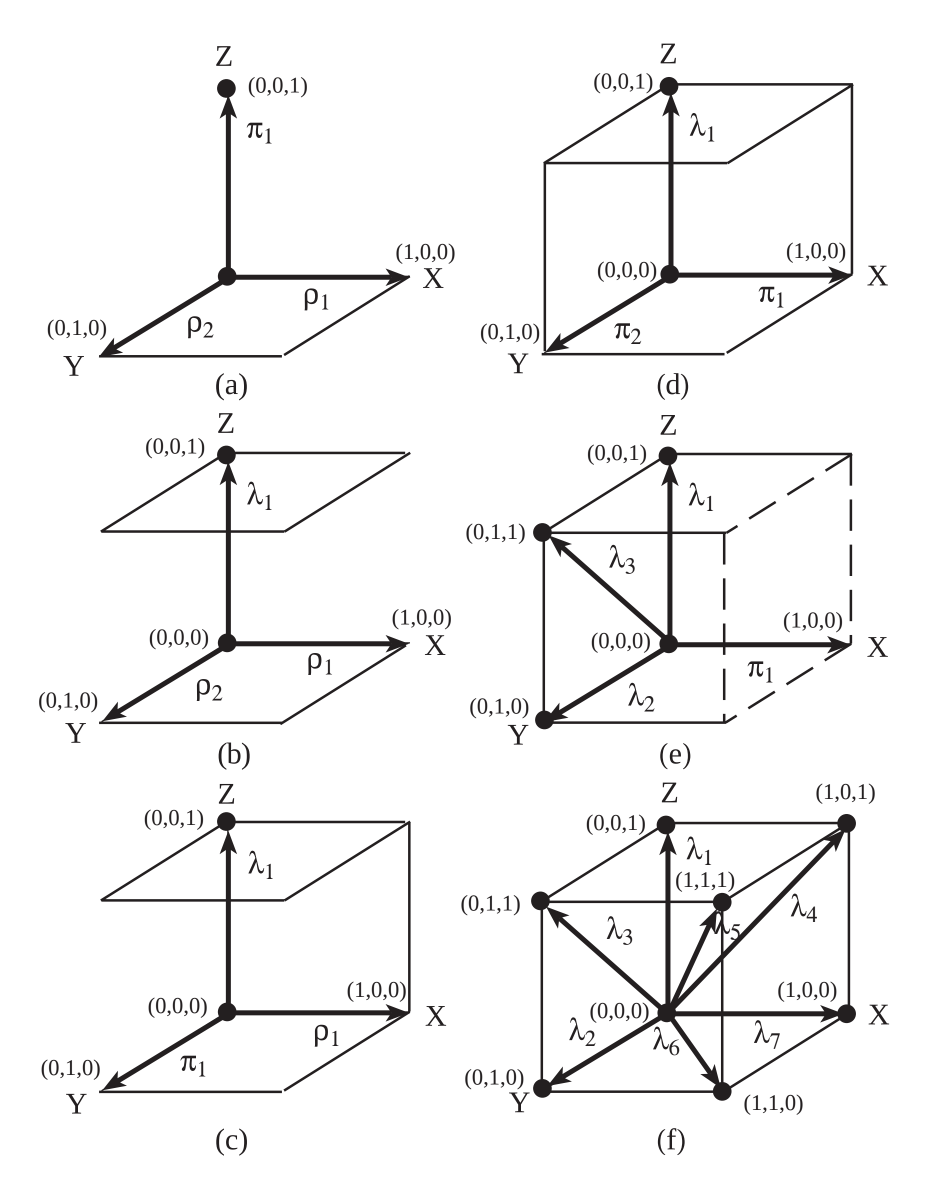


Figura 5.1: Ilustración de cómo evoluciona la solución a medida que se añaden más restricciones.

a continuación sus puntos extremos y sus direcciones extremas, conceptos de relevancia especial en el análisis de los conjuntos convexos.

Los conjuntos convexos son los conjuntos más sencillos que aparecen de forma natural en la programación matemática. Un conjunto S es convexo si la línea que une dos puntos arbitrarios de ese conjunto, pertenece al conjunto. La figura 5.2 muestra un conjunto convexo y otro que no lo es en $\mathbb{R}^2$. Una definición

Tabla 5.2: Conjunto de restricciones y soluciones correspondientes [1]

Sistema	Solución	figura 5.1
$z \geq 0$	$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \pi_1 \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} + \rho_1 \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} + \rho_2 \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$	(a)
$z \geq 0$ $z \leq 1$	$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \lambda_1 \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} + \rho_1 \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} + \rho_2 \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$	(b)
$z \geq 0$ $z \leq 1$ $y \geq 0$	$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \lambda_1 \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} + \rho_1 \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} + \pi_1 \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$	(c)
$z \geq 0$ $z \leq 1$ $y \geq 0$ $x \geq 0$	$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \lambda_1 \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} + \pi_1 \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} + \pi_2 \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$	(d)
$z \geq 0$ $z \leq 1$ $y \geq 0$ $x \geq 0$ $y \leq 1$	$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \lambda_1 \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} + \pi_1 \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} + \lambda_2 \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} + \lambda_3 \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix}$	(e)
$z \geq 0$ $z \leq 1$ $y \geq 0$ $x \geq 0$ $y \leq 1$ $x \leq 1$	$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \lambda_1 \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} + \lambda_2 \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} + \lambda_3 \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix} + \lambda_4 \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}$ $+ \lambda_5 \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} + \lambda_6 \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix} + \lambda_7 \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$	(f)

[1] Los coeficientes ρ, π, y λ son coeficientes sin restricciones de signo, no negativos y no negativos que suman 1. Los vectores asociados a los coeficientes ρ, π, y λ definen el espacio vectorial, el cono y el politopo, respectivamente, que conjuntamente definen las soluciones.

precisa de conjunto convexo se establece a continuación.

Definición 5.1 (conjunto convexo). *Un conjunto S en $\mathbb{R}^n$ se dice que es*

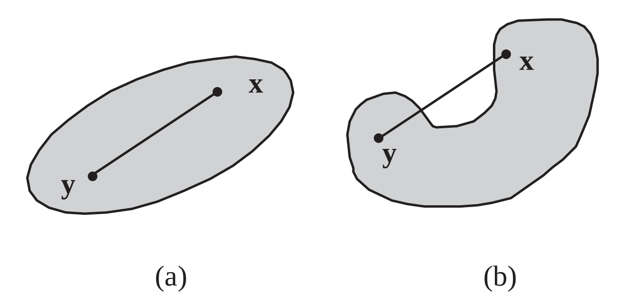


Figura 5.2: Conjunto convexo (a) y no convexo (b) en $\mathbb{R}^2$.

convexo si y solo si

$$\lambda \mathbf{x} + (1-\lambda)\mathbf{y} \in S \tag{5.2}$$

para todo $\lambda \in [0,1]$ *y* $\mathbf{x}, \mathbf{y} \in S$. ■

A continuación se muestran algunos ejemplos de conjuntos convexos.

1. (**Hiperplanos.**) Los hiperplanos son conjuntos definidos mediante

$$S = \left\{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{p}^T\mathbf{x} = \alpha\right\} \tag{5.3}$$

donde $\mathbf{p}$ es un vector no nulo en $\mathbb{R}^n$ y α un escalar. Empleando la definición de conjunto convexo, se demuestra fácilmente que S es un conjunto convexo. Tomando $\mathbf{x}, \mathbf{y} \in S$, por la definición de S, se obtiene que $\mathbf{p}^T\mathbf{x} = \alpha$ y $\mathbf{p}^T\mathbf{y} = \alpha$. Para cualquier $\lambda \in [0,1]$, se obtiene

$$\mathbf{p}^T\left[\lambda\mathbf{x} + (1-\lambda)\mathbf{y}\right] = \lambda\mathbf{p}^T\mathbf{x} + (1-\lambda)\mathbf{p}^T\mathbf{y} = \lambda\alpha + (1-\lambda)\alpha = \alpha$$

Por tanto, cualquier combinación convexa de vectores en S pertenece a S.

2. (**Semiespacios.**) La figura 5.3 muestra que el hiperplano

$$S = \{\mathbf{x} \in \mathbb{R}^n \mid x_1 + x_2 = 1\}$$

separa $\mathbb{R}^2$ en dos partes. En una de ellas, todos los puntos $\mathbf{x} = (x_1, x_2)$ satisfacen $x_1 + x_2 > 1$, y en la otra satisfacen $x_1 + x_2 < 1$. Los conjuntos $S^+ = \left\{\mathbf{x} \in \mathbb{R}^2 \mid x_1 + x_2 \geq 1\right\}$ y $S^- = \left\{\mathbf{x} \in \mathbb{R}^2 \mid x_1 + x_2 \leq 1\right\}$ se denominan *semiespacios*. En general, si S es el hiperplano determinado por (5.3), se definen los subespacios asociados a S, como los conjuntos

$$S^+ = \left\{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{q}^T\mathbf{x} \geq \alpha\right\} \quad \text{y } S^- = \left\{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{q}^T\mathbf{x} \leq \alpha\right\}$$

3. (**Conos convexos.**) Un conjunto C no vacío es un *cono* si $\mathbf{x} \in C$ implica que $\lambda\mathbf{x} \in C$ para todo $\lambda > 0$. Si, además, C es convexo, se dice que es un *cono convexo*. La figura 5.4 muestra ejemplos de conos convexos y no convexos.

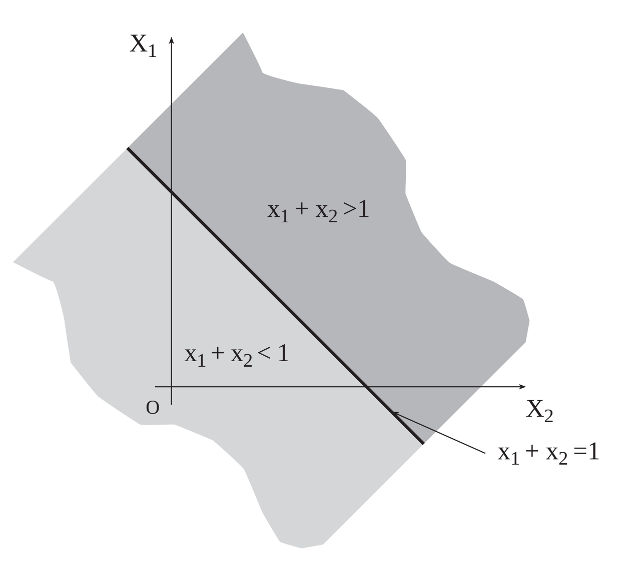


Figura 5.3: Dos regiones separadas por un hiperplano.

Se invita al lector a probar la convexidad en los casos 2 y 3.

Las propiedades siguientes son consecuencia directa de la definición de convexidad:

- Si $\{S_i\}_{i \in I}$ es una familia de conjuntos convexos, entonces, el conjunto

  $$\cap_{i \in I} S_i \equiv \{\mathbf{x} \mid \mathbf{x} \in S_i \text{ par todo } i \in I\}$$

  es también un conjunto convexo (véase la figura 5.5).

- Si S_1 y S_2 son conjuntos convexos, el conjunto

  $$S_1 + S_2 \equiv \{\mathbf{y} = \mathbf{x}_1 + \mathbf{x}_2 \mid \mathbf{x}_1 \in S_1 \text{ y } \mathbf{x}_2 \in S_2\}$$

  también lo es.

Estas propiedades pueden emplearse para definir nuevos conjuntos convexos mediante el empleo de subespacios. Se puede considerar la intersección, finita o infinita, de subespacios, y el conjunto resultante es convexo.

A continuación se tratan los conceptos de puntos extremos y direcciones extremas en los conjuntos convexos.

Definición 5.2 (punto extremo). *Sea S un conjunto convexo no vacío en $\mathbb{R}^n$. Un vector $\mathbf{x} \in S$ se denomina punto extremo de S si $\mathbf{x} = \lambda \mathbf{x}_1 + (1-\lambda)\mathbf{x}_2$ con $\mathbf{x}_1, \mathbf{x}_2 \in S$ y $\lambda \in (0,1)$ implica que $\mathbf{x} = \mathbf{x}_1 = \mathbf{x}_2$.* ∎

En un círculo el número de puntos extremos es infinito pero en un triángulo o en un cuadrado, éste número es finito (véase la figura 5.6). Los siguientes conjuntos E son conjuntos de puntos extremos de determinados conjuntos convexos S:

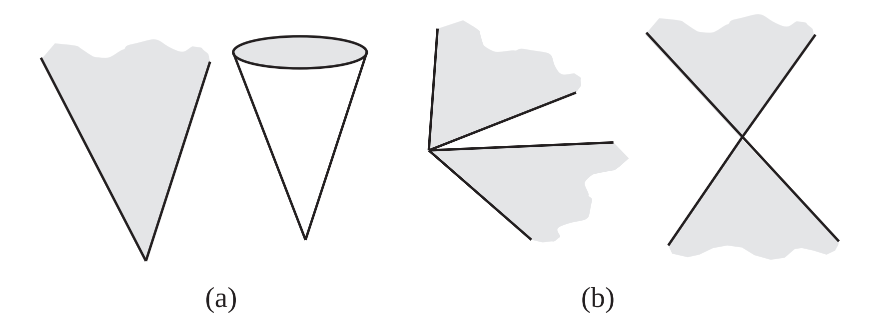


Figura 5.4: Cono convexo (a) y no convexo (b).

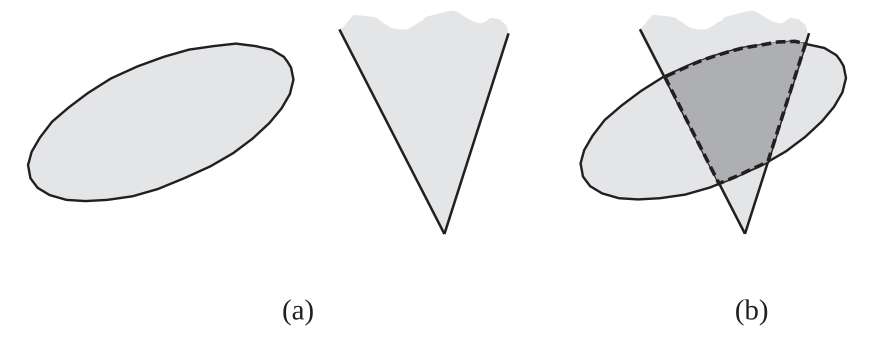


Figura 5.5: Conjuntos convexos (a) como intersección de (b) conjuntos convexos.

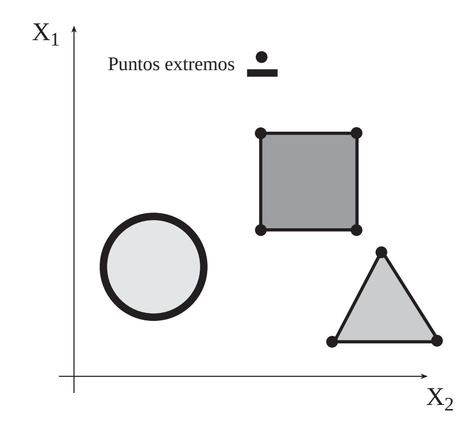


Figura 5.6: Puntos extremos en un círculo, un cuadrado y un triángulo.

1. Círculo

$$
\begin{aligned}
S &= \left\{(x_1, x_2) \in \mathbb{R}^2 \mid x_1^2 + x_2^2 \leq 1\right\} \\
E &= \left\{(x_1, x_2) \in \mathbb{R}^2 \mid x_1^2 + x_2^2 = 1\right\}
\end{aligned}
$$

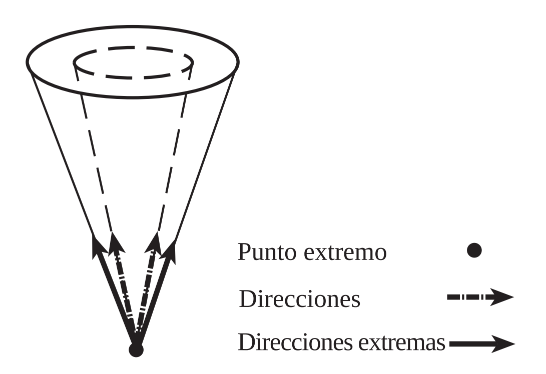


Figura 5.7: Cono: dos de sus infinitas direcciones y direcciones extremas, y su único punto extremo.

2. Cuadrado

$$S = \left\{(x_1, x_2) \in \mathbb{R}^2 \mid x_1 \geq 0,\ x_2 \geq 0,\ x_1 \leq 1,\ x_2 \leq 1\right\}$$

$$E = \{(0,0),\ (0,1),\ (1,1),\ (1,0)\}$$

3. Triángulo

$$S = \left\{(x_1, x_2) \in \mathbb{R}^2 \mid x_1 \geq 0,\ x_1 - x_2 \geq 0,\ x_1 + x_2 \leq 1\right\}$$

$$E = \left\{(0,0),\ (\tfrac{1}{2}, \tfrac{1}{2}),\ (0,1)\right\}$$

Teorema 5.1 (teorema de representación de conjuntos convexos finitos). *Si un conjunto convexo está acotado y cerrado, cualquiera de sus puntos puede escribirse como una combinación convexa de sus puntos extremos.* ■

El Teorema 5.1 no es cierto para un conjunto convexo no acotado. Por ejemplo $C = \left\{\mathbf{x} \in \mathbb{R}^3 \mid x_1^2 + x_2^2 \leq x_3^2,\ x_3 \geq 0\right\}$, que se muestra en la figura 5.7. Obsérvese que C es convexo y cerrado. Sin embargo, S contiene exclusivamente el punto extremo $(0,0,0)$, es decir, el origen, y ciertamente C no es igual al conjunto de todas las combinaciones convexas de sus puntos extremos. Esto ocurre porque C no está acotado. Para tratar conjuntos no acotados se necesita el concepto de dirección extrema.

Definición 5.3 (dirección). *Sea S un conjunto convexo, cerrado y no vacío en $\mathbb{R}^n$. Se dice que un vector unidad $\mathbf{d}$ es una dirección de S, si para cada $\mathbf{x} \in S$, $\mathbf{x} + \pi\mathbf{d} \in S$ para todo $\pi \geq 0$.* ■

Definición 5.4 (dirección extrema). *Una dirección $\mathbf{d}$ se denomina extrema si no puede expresarse como una combinación lineal positiva de dos direcciones distintas, esto es, si $\mathbf{d} = \pi_1\mathbf{d}_1 + \pi_2\mathbf{d}_2$ para $\pi_1,\ \pi_2 > 0$, entonces $\mathbf{d} = \mathbf{d}_1 = \mathbf{d}_2$.* ■

Para ilustrar estas definiciones, considérese el conjunto C que se muestra en la figura 5.7. El conjunto de direcciones es

$$D = \left\{ \mathbf{d} \in \mathbb{R}^3 \mid d_1^2 + d_2^2 \leq d_3^2,\ d_1^2 + d_2^2 + d_3^2 = 1 \right\},$$

y el de direcciones extremas $D_E = \left\{ \mathbf{d} \in \mathbb{R}^3 \mid d_1^2 + d_2^2 = d_3^2 = 1/2 \right\}$.

En las siguientes secciones se describen casos particulares de conjuntos convexos.

5.3 Espacios vectoriales

Los espacios vectoriales son la clase más conocida de conjuntos convexos. Surgen como resultado de la combinación lineal de un conjunto de vectores.

Definición 5.5 (combinación lineal). *Considérese una matriz* $\mathbf{A}$*, y el conjunto de sus vectores columna*

$$\mathbf{A} \equiv \{\mathbf{a}_1, \ldots, \mathbf{a}_k\}$$

Un vector $\mathbf{x}$ *se dice que es una combinación lineal de los vectores del conjunto* $\mathbf{A}$ *si puede expresarse como*

$$\mathbf{x} = \rho_1 \mathbf{a}_1 + \cdots + \rho_k \mathbf{a}_k \tag{5.4}$$

donde $\rho_i \in \mathbb{R}, \forall i$. ■

En lo que sigue, para simplificar la notación, se identifica el conjunto $\mathbf{A}$ con la matriz $\mathbf{A}$. Al conjunto de todas las posibles combinaciones de los vectores de $\mathbf{A}$ se le denomina $\mathbf{A}_\rho$.

Definición 5.6 (espacio vectorial). *El conjunto*

$$\mathbf{A}_\rho \equiv \{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{x} = \rho_1 \mathbf{a}_1 + \ldots + \rho_k \mathbf{a}_k \quad con \quad \rho_i \in \mathbb{R}; i = 1, \ldots, k\}$$

de todos los vectores del espacio euclídeo $\mathbb{R}^n$*, que pueden generarse como combinaciones lineales de los vectores del conjunto* $\mathbf{A}$ *se denomina el espacio vectorial generado por* $\mathbf{A}$*, y el conjunto* $\mathbf{A}$ *es el conjunto de sus generadores.* ■

Es de interés conocer si un subconjunto de un espacio vectorial es un subespacio vectorial. A este efecto, se proporciona sin demostración el siguiente teorema.

Teorema 5.2 (identificación de subespacios). *Sea* $\mathcal{V}$ *un espacio vectorial y* $\mathcal{W}$ *un subespacio de él. Entonces,* $\mathcal{W}$ *es un subespacio vectorial de* $\mathcal{V}$ *si y sólo si*

1. $\mathbf{a}, \mathbf{b} \in \mathcal{W} \Rightarrow \mathbf{a} + \mathbf{b} \in \mathcal{W}$
2. $\mathbf{a} \in \mathcal{W}, \alpha \in \mathbb{R} \Rightarrow \alpha \mathbf{a} \in \mathcal{W}$

■

La demostración de este teorema se puede encontrar en Castillo et al. [21]. Nótese que la condición 2 prueba que un subespacio vectorial es también un cono.

El teorema siguiente muestra que el conjunto de todas las soluciones factibles de un sistema homogéneo de ecuaciones lineales es un espacio vectorial, y que cualquier espacio vectorial puede escribirse como un sistema homogéneo de ecuaciones lineales.

Teorema 5.3 (representación de un espacio vectorial). *Sea S un subconjunto de $\mathbb{R}^n$. El conjunto S es un subespacio vectorial de $\mathbb{R}^n$ si y sólo si existe una matriz $\mathbf{H}$ de dimensión $m \times n$, tal que*

$$S = \{\mathbf{x} \in \mathbb{R}^n | \mathbf{H}\mathbf{x} = \mathbf{0}\} \tag{5.5}$$

■

Ejemplo 5.1 (conjunto factible de soluciones de un sistema homogéneo de ecuaciones lineales). La solución general del sistema de ecuaciones lineales:

$$\begin{array}{cccccc} x_1 & -x_2 & & -x_4 & = & 0 \\ x_1 & & -x_3 & +x_4 & = & 0 \\ & x_2 & -x_3 & +2x_4 & = & 0 \end{array} \tag{5.6}$$

es un subespacio vectorial

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \rho_1 \begin{pmatrix} 1 \\ 1 \\ 1 \\ 0 \end{pmatrix} + \rho_2 \begin{pmatrix} -1 \\ -2 \\ 0 \\ 1 \end{pmatrix}; \quad \rho_1, \rho_2 \in \mathbb{R} \tag{5.7}$$

que puede obtenerse empleando herramientas de software tales como Mathematica [103], por ejemplo. Obsérvese que existen 2 grados de libertad (ρ_1 y ρ_2) para generar las soluciones de (5.6). ■

5.4 Conos poliédricos convexos

En esta sección se definen los conos poliédricos convexos, que se caracterizan por obtenerse mediante combinaciones lineales no negativas de vectores.

Definición 5.7 (combinación lineal no negativa). *Se dice que un vector $\mathbf{x}$ es una combinación lineal no negativa de los vectores $\mathbf{A} = \{\mathbf{a}_1, \ldots, \mathbf{a}_k\}$ si y sólo si*

$$\mathbf{x} = \pi_1 \mathbf{a}_1 + \cdots + \pi_k \mathbf{a}_k \tag{5.8}$$

donde $\pi_i \in \mathbb{R}^+, \forall i$, $\mathbb{R}^+$ es un conjunto de números reales no negativos. ■

Al conjunto de todas las combinaciones lineales no negativas del conjunto de vectores de $\mathbf{A}$ se le denomina $\mathbf{A}_\pi$.

Definición 5.8 (cono poliédrico convexo). *Sea* $\mathbf{A} = \{\mathbf{a}_1, \ldots, \mathbf{a}_k\}$. *El conjunto*

$$\mathbf{A}_\pi \equiv \{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{x} = \pi_1 \mathbf{a}_1 + \ldots + \pi_k \mathbf{a}_k \quad con \quad \pi_j \geq 0; j = 1, \ldots, k\}$$

de todas las combinaciones lineales no negativas de los vectores de $\mathbf{A}$ *se denomina cono poliédrico convexo. Los vectores* $\mathbf{a}_1, \ldots, \mathbf{a}_k$ *son los generadores del cono.* ■

En lo que sigue y por simplicidad se denominará a los conos poliédricos convexos simplemente conos. Por tanto, el conjunto C es un cono si y sólo si existe un conjunto de vectores $\mathbf{A}$ tales que $C = \mathbf{A}_\pi$.

Nota 5.1 *Obsérvese que "conos convexos" y "conos poliédricos convexos" son conceptos diferentes.* ■

Nota 5.2 *Nótese que los generadores del cono son a su vez direcciones del cono.* ■

Un cono C que no es un subespacio vectorial es el resultado del empleo de variables restringidas en signo (no negativas o no positivas).

Definición 5.9 (forma general del cono poliédrico convexo). *Los generadores* $\mathbf{A} = (\mathbf{a}_1, \ldots, \mathbf{a}_m)$ *del cono* $\mathbf{A}_\pi$, *pueden clasificarse en dos grupos:*

1. *Generadores cuyos vectores opuestos pertenecen al cono:*

$$\mathbf{B} \equiv \{\mathbf{a}_i \mid -\mathbf{a}_i \in \mathbf{A}_\pi\}$$

2. *Generadores cuyos vectores opuestos no pertenecen al cono:*

$$\mathbf{C} \equiv \{\mathbf{a}_i \mid -\mathbf{a}_i \notin \mathbf{A}_\pi\}$$

Por tanto, el cono puede expresarse de la manera siguiente:

$$\mathbf{A}_\pi \equiv (\mathbf{B} : -\mathbf{B} : \mathbf{C})_\pi \equiv \mathbf{B}_\rho + \mathbf{C}_\pi \tag{5.9}$$

que se conoce como la forma general de un cono. ■

El teorema siguiente muestra que el conjunto de todas las soluciones factibles de un sistema homogéneo de desigualdades lineales es un cono, y que todo cono puede expresarse como un sistema homogéneo de desigualdades lineales.

Teorema 5.4 (representación de un cono). *Sea S un subconjunto de $\mathbb{R}^n$. El conjunto S es un cono con origen en $\mathbf{0}$ si y sólo si existe una matriz $\mathbf{H}$ de dimensión $m \times n$, tal que*

$$S = \{\mathbf{x} \in \mathbb{R}^n | \mathbf{H}\mathbf{x} \leq \mathbf{0}\} \tag{5.10}$$

■

Ejemplo 5.2 (conjunto de soluciones factibles de un sistema homogéneo lineal de desigualdades). La solución general del sistema lineal de desigualdades:

$$\begin{array}{cccccc} x_1 & -x_2 & & -x_4 & \leq & 0 \\ x_1 & & -x_3 & +x_4 & \leq & 0 \\ & x_2 & -x_3 & +2x_4 & \leq & 0 \end{array} \tag{5.11}$$

es un cono

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \rho_1 \begin{pmatrix} 1 \\ 1 \\ 1 \\ 0 \end{pmatrix} + \rho_2 \begin{pmatrix} -1 \\ -2 \\ 0 \\ 1 \end{pmatrix} + \pi_1 \begin{pmatrix} -1 \\ -1 \\ 0 \\ 0 \end{pmatrix} + \pi_2 \begin{pmatrix} -1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \tag{5.12}$$
$$\rho_1, \rho_2 \in \mathbb{R}; \pi_1, \pi_2 \in \mathbb{R}^+$$

La obtención de los generadores de un cono es compleja pero puede hacerse mediante el empleo del cono dual (véase el Apéndice A).

Se dispone de 4 grados de libertad (ρ_1, ρ_2, π_1 y π_2) para generar las soluciones de (5.11); sin embargo, dos de los coeficientes (π_1 y π_2) deben ser no negativos. Obsérvese que un cono se expresa como la suma de un espacio vectorial y un cono propio (un cono cuya componente de espacio vectorial no existe), esto es, en su forma general.

■

Nota 5.3 *Un espacio vectorial $\mathbf{A}_\rho$ es un caso particular de cono:*

$$\mathbf{A}_\rho \equiv (\mathbf{A} : -\mathbf{A})_\pi, \tag{5.13}$$

donde $-\mathbf{A}$ es el negativo de $\mathbf{A}$. Lo anterior significa que un espacio vectorial es un cono generado por sus generadores y los vectores opuestos de estos generadores.

La Expresión (5.13) muestra que el concepto de cono es más amplio que el concepto de espacio vectorial y, lo que es más importante, permite el empleo de métodos específicos asociados a los conos, en vez de los métodos clásicos del álgebra lineal. Los métodos específicos asociados a conos son más eficientes ya que el concepto de cono es más amplio que el concepto de espacio vectorial. ■

5.5 Politopos

Definición 5.10 (combinación lineal convexa). *Se dice que un vector* $\mathbf{x}$ *es una combinación lineal convexa de los vectores de* $\mathbf{A} = \{\mathbf{a}_1, \ldots, \mathbf{a}_k\}$ *si y sólo si*

$$\mathbf{x} = \lambda_1 \mathbf{a}_1 + \cdots + \lambda_k \mathbf{a}_k, \tag{5.14}$$

donde $\lambda_i \in \mathbb{R}^+, \forall i$ *y* $\sum_{i=1}^{k} \lambda_i = 1.$ ∎

El conjunto de todas las combinaciones lineales convexas de los vectores de $\mathbf{A}$ se denomina $\mathbf{A}_\lambda$.

Definición 5.11 (politopo). *Sea* $\mathbf{A} = \{\mathbf{a}_1, \ldots, \mathbf{a}_k\}$. *El conjunto*

$$S = \mathbf{A}_\lambda \equiv \left\{ \mathbf{x} \in \mathbb{R}^n \mid \mathbf{x} = \lambda_1 \mathbf{p}_1 + \cdots + \lambda_k \mathbf{p}_k \quad con \quad \lambda_i \geq 0 \; ; \; \sum_{i=1}^{k} \lambda_i = 1 \right\},$$

de todas las combinaciones lineales convexas de los vectores de $\mathbf{A}$ *se denomina politopo o envoltura convexa generada por* $\{\mathbf{a}_1, \ldots, \mathbf{a}_k\}$. *Si el conjunto de vectores* $\{\mathbf{a}_1, \ldots, \mathbf{a}_k\}$ *es mínimo, se trata del conjunto de los* k *puntos extremos de* S. ∎

La figura 5.8 muestra dos ejemplos de politopos convexos. El caso (a) es un tetraedro dado que los puntos asociados con los generadores $\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3$, y $\mathbf{a}_4$ no están en el mismo plano. Sin embargo, en el caso (b), los puntos están en el mismo plano y el politopo degenera en un cuadrilátero.

El teorema siguiente muestra que cualquier politopo se puede expresar como un sistema lineal completo de desigualdades.

Teorema 5.5 (representación de un politopo). *Un politopo* S *puede expresarse como la intersección de un conjunto finito de semiespacios:*

$$S = \{\mathbf{x} \in \mathbb{R}^n | \mathbf{H}\mathbf{x} \leq \mathbf{b}\} \tag{5.15}$$

∎

Ejemplo 5.3 (representación de un politopo). Considérese el politopo definido por el conjunto de desigualdades

$$\begin{array}{rrcl} -x_1 & -x_2 & \leq & 0 \\ x_1 & -x_2 & \leq & 1 \\ -x_1 & +x_2 & \leq & 0 \\ x_1 & +x_2 & \leq & 1 \end{array} \tag{5.16}$$

que se ilustra en la figura 5.9. El conjunto de sus puntos extremos es

$$\mathbf{P} = \left\{ (0,0)^T, \left(\frac{1}{2}, \frac{1}{2}\right)^T, (1,0)^T, \left(\frac{1}{2}, -\frac{1}{2}\right)^T \right\}.$$

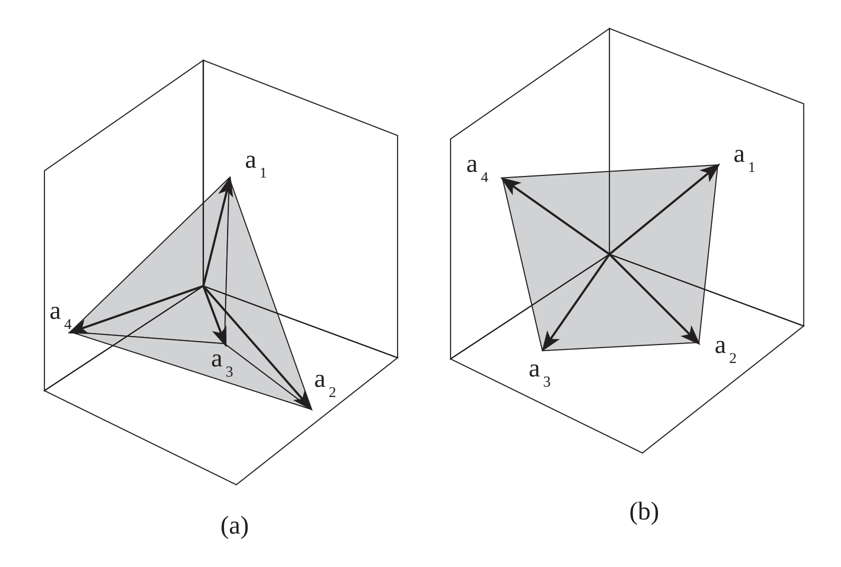


Figura 5.8: Dos ejemplos de politopos convexos.

Por tanto, S se puede representar mediante

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \lambda_1 \begin{pmatrix} 0 \\ 0 \end{pmatrix} + \lambda_2 \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix} + \lambda_3 \begin{pmatrix} 1 \\ 0 \end{pmatrix} + \lambda_4 \begin{pmatrix} \frac{1}{2} \\ -\frac{1}{2} \end{pmatrix} \tag{5.17}$$

donde

$$\begin{array}{ccccccc} \lambda_1 & +\lambda_2 & +\lambda_3 & +\lambda_4 & = & 1 \\ \lambda_1 & & & & \geq & 0 \\ & \lambda_2 & & & \geq & 0 \\ & & \lambda_3 & & \geq & 0 \\ & & & \lambda_4 & \geq & 0 \end{array} \tag{5.18}$$

■

5.6 Poliedro

Esta sección se dedica a los poliedros.

Definición 5.12 (poliedro). *Un poliedro es la intersección de un número finito de semiespacios:*

$$S = \{\mathbf{x} \in \mathbb{R}^n | \mathbf{Hx} \leq \mathbf{b}\} \tag{5.19}$$

Si S está acotado, S es un politopo. ■

La expresión (5.19) muestra que el conjunto de todas las soluciones factibles de un conjunto de desigualdades es un poliedro.

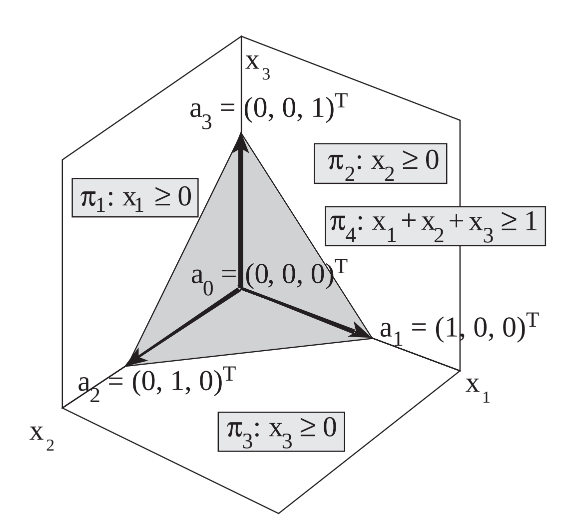


Figura 5.9: Dos definiciones diferentes de politopo: mediante sus puntos extremos y mediante un conjunto de desigualdades.

Ejemplo 5.4 (representación de conjuntos poliédricos). La solución del sistema lineal de desigualdades

$$\begin{array}{cccccc} x_1 & -x_2 & & -x_4 & \leq & 3 \\ x_1 & & -x_3 & +x_4 & \leq & 1 \\ & x_2 & -x_3 & +2x_4 & \leq & 0 \end{array} \tag{5.20}$$

puede obtenerse utilizando el algoritmo Γ (véase la Sección A.1), y es el poliedro

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \lambda_1 \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} + \lambda_2 \begin{pmatrix} 1 \\ -2 \\ 0 \\ 0 \end{pmatrix} + \pi_1 \begin{pmatrix} -1 \\ -1 \\ 0 \\ 0 \end{pmatrix} + \pi_2 \begin{pmatrix} -1 \\ 0 \\ 0 \\ 0 \end{pmatrix} + \pi_3 \begin{pmatrix} -1 \\ -2 \\ 0 \\ 1 \end{pmatrix} + \pi_4 \begin{pmatrix} 1 \\ 2 \\ 0 \\ -1 \end{pmatrix} \tag{5.21}$$

donde

$$\begin{array}{rcl} \lambda_1 + \lambda_2 & = & 1 \\ \lambda_1,\ \lambda_2 & \geq & 0 \\ \pi_1,\ \pi_2,\ \pi_3,\ \pi_4 & \geq & 0 \end{array} \tag{5.22}$$

Esta representación tiene 6 grados de libertad (λ_1, λ_2, π_1, π_2, π_3, y π_4) en lo que se refiere a generar soluciones de (5.20). Obsérvese que un poliedro se expresa como la suma de un politopo y un cono. ∎

El conjunto de restricciones de un PPL define un poliedro. Casos particulares de poliedros son los espacios vectoriales, los conos y los politopos (véase la figura 5.10). El teorema siguiente muestra que cualquier poliedro puede escribirse como suma de estas tres estructuras básicas.

Teorema 5.6 (representación general de los poliedros). *Cualquier poliedro S se puede expresar como la suma de un espacio vectorial, un cono y un politopo:*

$$S = \mathbf{V}_\rho + \mathbf{W}_\pi + \mathbf{Q}_\lambda.$$

Esto quiere decir que $\mathbf{x} \in S$ si y sólo si $\mathbf{x}$ se puede expresar como

$$\mathbf{x} = \sum_i \rho_i \mathbf{v}_i + \sum_j \pi_j \mathbf{w}_j + \sum_k \lambda_k \mathbf{q}_k;\ \rho_i \in \mathbb{R};\ \pi_j \geq 0;\ \lambda_k \geq 0;\ \sum_k \lambda_k = 1 \tag{5.23}$$

Nótese que alguno de estos sumandos puede no existir. ■

Corolario 5.1 *Si el espacio vectorial no existe, el conjunto mínimo de generadores del cono y del politopo está compuesto por los conjuntos de direcciones extremas y de puntos extremos, respectivamente. En este caso, el conjunto de direcciones extremas es $\mathbf{W}$, y el conjunto de puntos extremos es $\mathbf{Q}$.*

Los resultados anteriores se resumen en la Tabla 5.1 y en la figura 5.10.

5.7 PPL acotado y no acotado

Este capítulo concluye mostrando la relación entre la estructura de la región factible de un PPL y su carácter acotado o no acotado.

Considérese el problema siguiente. Minimizar

$$Z = \mathbf{c}^T \mathbf{x}$$

sujeto a una de las restricciones de la Tabla 5.1. De acuerdo con esta tabla, se pueden dar los casos siguientes:

Espacio vectorial. En este caso el valor de la función objetivo es

$$\mathbf{c}^T \mathbf{x} = \sum_i \rho_i \mathbf{c}^T \mathbf{v}_i$$

que da lugar a un problema acotado si $\mathbf{c}^T$ es ortogonal a todos los vectores $\mathbf{v}_i$. En otro caso, el problema no está acotado puesto que los valores de ρ_i pueden elegirse tan grandes como se quiera.

Cono. En este caso el valor de la función objetivo es

$$\mathbf{c}^T \mathbf{x} = \sum_j \pi_j \mathbf{c}^T \mathbf{w}_i;\ \pi_j \geq 0$$

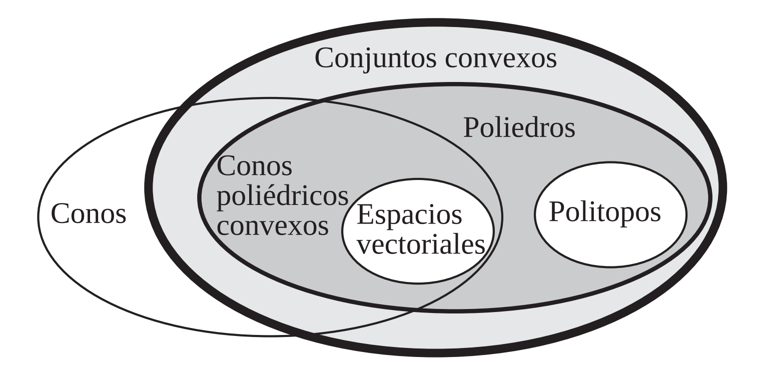


Figura 5.10: La diferentes estructuras que surgen de la resolución de un sistema lineal de igualdades y desigualdades.

que da lugar a un problema acotado solo si $\mathbf{c}^T$ es ortogonal a todos los vectores $\mathbf{w}_j$; es un problema acotado por abajo solo si $\mathbf{c}^T\mathbf{w}_j \geq 0;\ \forall j$, y por arriba, si $\mathbf{c}^T\mathbf{w}_j \leq 0;\ \forall j$. En cualquier otro caso, el problema es no acotado.

Politopo. En este caso el valor de la función objetivo es

$$\mathbf{c}^T\mathbf{x} = \sum_k \lambda_k \mathbf{c}^T\mathbf{q}_k;\ \lambda_k \geq 0$$

que siempre da lugar a un problema acotado.

Poliedro. En este caso el valor de la función objetivo es

$$\mathbf{c}^T\mathbf{x} = \sum_i \rho_i \mathbf{v}_i + \sum_j \pi_j \mathbf{w}_j + \sum_k \lambda_k \mathbf{c}^T\mathbf{q}_k$$

lo que da lugar a los siguientes casos:

1. Un problema es acotado solamente si $\mathbf{c}^T$ es ortogonal a todos los vectores $\mathbf{v}_i$ y $\mathbf{w}_j$
2. Un problema es acotado inferiormente solamente si $\mathbf{c}^T$ es ortogonal a todos los vectores $\mathbf{v}_i$ y $\mathbf{c}^T\mathbf{w}_j \geq 0;\ \forall j$
3. Un problema es acotado superiormente solamente si $\mathbf{c}^T$ es ortogonal a todos los vectores $\mathbf{v}_i$ y $\mathbf{c}^T\mathbf{w}_j \leq 0;\ \forall j$
4. Un problema es no acotado en cualquier otro caso.

Ejercicios

5.1 Demuéstrese que el conjunto $S = \{\mathbf{x} \in \mathbb{R}^3 \mid x_1^2 + x_2^2 \leq x_3^2\}$ es un cono.

5.2 Dados los siguientes conjuntos poliédricos

$$\begin{array}{l} S_1 = \left\{\mathbf{x} \in \mathbb{R}^2 \mid x_1 + x_2 \leq 1,\ x_1 \geq 0,\ x_2 \geq 0\right\} \\ S_2 = \left\{\mathbf{x} \in \mathbb{R}^2 \mid x_1 - x_2 \leq 0,\ x_1 \geq 0,\ x_2 \geq 0\right\} \\ S_3 = \left\{\mathbf{x} \in \mathbb{R}^2 \mid x_1 - x_2 \leq 0,\ x_1 + x_2 \leq 1,\ x_1 \geq 0,\ x_2 \geq 0\right\} \\ S_4 = \left\{\mathbf{x} \in \mathbb{R}^2 \mid x_2 - x_1 \leq 0,\ x_1 - x_2 \leq 0,\ x_1 + x_2 \leq 1,\ x_1 \geq 0,\ x_2 \geq 0\right\} \\ S_5 = \left\{\mathbf{x} \in \mathbb{R}^2 \mid x_1 + x_2 \leq 1,\ 2x_1 + 2x_2 \geq 3\right\} \end{array}$$

(a) Represéntense estos conjuntos empleando puntos extremos y direcciones extremas.

(b) Dibújense los conjuntos anteriores e indíquese en la figura los puntos extremos y las direcciones extremas.

5.3 ¿Por qué no es posible representar el conjunto $S = \{\mathbf{x} \in \mathbb{R}^3 \mid x_1 \geq 0\}$ empleando exclusivamente puntos extremos y direcciones extremas?

5.4 Sea $\Phi(\mathbf{x}) = \mathbf{A}\mathbf{x}$ una función lineal, donde $\mathbf{A}$ es una matriz de dimensión $m \times n$. Demuéstrese que si S es un conjunto poliédrico acotado en $\mathbb{R}^n$, entonces $\Phi(S)$ es un conjunto poliédrico acotado en $\mathbb{R}^m$.

5.5 Si S_1 y S_2 son conjuntos poliédricos en $\mathbb{R}^n$, entonces $S_1 + S_2$ es también un conjunto poliédrico. (*Ayuda*: Empléese el resultado anterior.)

5.6 Considérese el conjunto siguiente de restricciones:

(a)

$$\begin{array}{rrcl} -2x_1 & +\, x_2 & = & 0 \end{array}$$

(b)

$$\begin{array}{rrcl} -2x_1 & +\, x_2 & \leq & 0 \\ & -\, x_2 & \leq & 0 \end{array}$$

(c)

$$\begin{array}{rrcl} -2x_1 & +\, x_2 & \leq & 0 \\ & -\, x_2 & \leq & 0 \\ x_1 & -\, x_2 & \leq & 2 \end{array}$$

(d)

$$\begin{array}{rrcl} -2x_1 & +\, x_2 & \leq & 0 \\ & -\, x_2 & \leq & 0 \\ x_1 & -\, x_2 & \leq & 2 \\ x_1 & +\, x_2 & \leq & 6 \end{array}$$

(i) Dibújense los correspondiente conjuntos y señálense los puntos extremos y las direcciones extremas en las figuras.

(ii) Escríbanse expresiones para esos conjuntos en función de sus puntos extremos y direcciones extremas.

(iii) Identifíquense los conjuntos anteriores como espacios vectoriales, conos, politopos o poliedros.

(iv) Obténgase gráficamente los valores máximo y mínimo de la función objetivo $Z = x_1 + x_2$.

(v) Empleando los conjuntos anteriores, propóngase un PPL para el que existan soluciones múltiples.

5.7 Considérese el conjunto de restricciones siguientes:

(a)

$$\begin{array}{rcl} -x_1 \quad + x_3 & = & 0 \end{array}$$

(b)

$$\begin{array}{rrrcl} -x_1 & & + x_3 & \leq & 0 \\ & & - x_3 & \leq & 0 \\ & -x_2 & + x_3 & \leq & 0 \end{array}$$

(c)

$$\begin{array}{rrrcl} -x_1 & & + x_3 & \leq & 0 \\ & & - x_3 & \leq & 0 \\ & -x_2 & + x_3 & \leq & 0 \\ x_1 & +x_2 & & \leq & 2 \end{array}$$

(i) Dibújense los correspondiente conjuntos y señálense los puntos extremos y las direcciones extremas en las figuras.

(ii) Escríbanse expresiones para esos conjuntos en función de sus puntos extremos y direcciones extremas.

(iii) Identifíquense los conjuntos anteriores como espacios vectoriales, conos, politopos o poliedros.

(iv) Obténgase gráficamente los valores máximo y mínimo de la función objetivo $Z = x_1 + x_2$.

(v) Empleando los conjuntos anteriores, propóngase un PPL para el que existan soluciones múltiples.

Capítulo 6

Resolución de problemas de programación lineal

6.1 Introducción

En el Capítulo 5 se ha desarrollado un método para obtener el conjunto de todas las soluciones factibles, determinadas por el conjunto de restricciones, basado en el cálculo de todos los puntos extremos y direcciones extremas, utilizando el algoritmo Γ. No obstante, este método es costoso desde el punto de vista computacional y no demasiado adecuado para los PPL. En este capítulo se presentan otros métodos desarrollados especialmente para este tipo de problemas.

Los economistas de la antigua Unión Soviética fueron los primeros en aplicar las técnicas de la programación lineal en la organización y planificación de la producción. Sin embargo, fue durante la Segunda Guerra Mundial cuando la programación lineal adquirió importancia. La Fuerza Aérea de los Estados Unidos creó el proyecto SCOOP *(Scientific Computation of Optima Programs)* dirigido por G. B. Dantzig [30]. El método más conocido para resolver problemas de programación lineal, *el método simplex*, es debido a Dantzig, quien lo introdujo en 1947. Afortunadamente, el crecimiento de la capacidad de cálculo de los computadores ha permitido el uso de las técnicas desarrolladas en problemas de gran dimensión.

Durante las últimas décadas se ha dedicado mucho trabajo e investigación a los problemas de programación lineal (PPL). Mientras que la implementación del método simplex ha sufrido modificaciones importantes, los conceptos fundamentales no han variado. Una descripción de la implementación para computador de este método puede verse, por ejemplo, en la *IBM Optimization Subroutine Library* (OSL) [56] o en Kuenzi et al. [64].

Para muchos problemas de programación lineal, el método simplex sigue siendo el mejor. Sin embargo, se han introducido mejoras diversas, como el método simplex revisado, el dual, o los métodos primal–dual. Este último trabaja simultáneamente con el primal y el dual. Esto puede presentar ventajas

pues se puede explotar la bien conocida relación entre las formulaciones primal y dual de un determinado problema (véase, por ejemplo, Forrest y Tomlin [40]). Aún así, según se indica, por ejemplo, en Press et al. [89]:

> "There seems to be no clearcut evidence that these methods are superior to the usual method by any factor substantially larger than the 'tender-loving-care factor' which reflects the programming effort of the proponents."

Para la mayoría de los casos el primal es el método preferido; aunque, en algunos casos especiales, otros métodos pueden dar mejores resultados.

El método del punto interior (MPI) fue introducido por Karmarkar [57] y es una alternativa al método simplex (MS) (véase Gill et al. [48]). El método primal–dual ha sido superado por el método predictor–corrector, que es a su vez una versión modificada del primal–dual original de Mehrotra [75]. Este método requiere normalmente menos iteraciones y es el método de punto interior preferido en muchas ocasiones.

Al contrario del MS, que tiene complejidad computacional exponencial, el MPI tiene complejidad polinómica. En consecuencia, para problemas grandes (más de 2000 restricciones y variables) los métodos del punto interior superan al MS. No obstante, el MS encuentra una solución en un número finito de pasos, mientras que el MPI no siempre lo consigue, pues converge asintóticamente a la solución óptima. Si su robustez está en entredicho, la distinción indicada puede ser determinante.

Este capítulo se centra en métodos computacionales para resolver problemas de programación lineal. Se asume que el análisis de datos, variables, restricciones y función objetivo se ha llevado a cabo en una etapa anterior, y que se dispone de una formulación coherente y completa.

En lo que sigue se describe, en primer lugar, de un modo informal pero preciso, el método simplex (MS), después se describe el método del punto exterior (MPE), que es una implementación particular del simplex dual. El método del punto interior (MPI) se describirá en el capítulo 9.

6.2 El método simplex

Antes de empezar con la descripción del método simplex, se ilustra su idea básica mediante un ejemplo.

6.2.1 Ejemplo ilustrativo

Considérese el siguiente PPL en forma estándar. Minimizar

$$Z = -11 + x_2 + 6x_3 + 2x_4 - x_9 \tag{6.1}$$

sujeto a

$$\begin{array}{ccccccccccc}
x_1 & & -x_3 & +x_4 & & & & & & = & 6 \\
& x_2 & +3x_3 & & & & & & -x_9 & = & 4 \\
& & & & x_5 & & & & +x_9 & = & 3 \\
& & 2x_3 & -x_4 & & +x_6 & & & & = & 1 \\
& & & -2x_4 & & & +x_7 & & -x_9 & = & 2 \\
& & -x_3 & +x_4 & & & & +x_8 & -x_9 & = & 5
\end{array}$$

y

$$x_1, x_2, \ldots, x_9 \geq 0$$

Como se tienen 6 restricciones de igualdad, se pueden despejar 6 variables, por ejemplo, x_1, x_2, x_5, x_6, x_7, y x_8, como función del resto de variables:

$$\begin{array}{ccccccc}
x_1 & = & 6 & +x_3 & -x_4 & \\
x_2 & = & 4 & -3x_3 & & +x_9 \\
x_5 & = & 3 & & & -x_9 \\
x_6 & = & 1 & -2x_3 & +x_4 & \\
x_7 & = & 2 & & +2x_4 & +x_9 \\
x_8 & = & 5 & +x_3 & -x_4 & x_9
\end{array} \tag{6.2}$$

Esto supone dividir el conjunto de variables $\{x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9\}$ en dos conjuntos $\{x_1, x_2, x_5, x_6, x_7, x_8\}$ y $\{x_3, x_4, x_9\}$, que se conocen como variables *básicas* y *no básicas*, respectivamente.

A continuación, se sustituyen las variables básicas (6.2) en la función objetivo (6.1), en función de las variables no básicas, para obtener

$$Z = -7 + x_9 + 3x_3 + 2x_4.$$

De este modo, el PPL inicial es equivalente a un PPL en el que se minimiza

$$Z = -7 + x_9 + 3x_3 + 2x_4$$

sujeto a

$$\begin{array}{ccccccc}
x_1 & = & 6 & +x_3 & -x_4 & \\
x_2 & = & 4 & -3x_3 & & +x_9 \\
x_5 & = & 3 & & & -x_9 \\
x_6 & = & 1 & -2x_3 & +x_4 & \\
x_7 & = & 2 & & +2x_4 & +x_9 \\
x_8 & = & 5 & +x_3 & -x_4 & +x_9
\end{array}$$

y

$$x_1, x_2, \ldots, x_9 \geq 0$$

La solución de este problema es $z = -7$ y se alcanza en el punto

$$(x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9) = (6, 4, 0, 0, 3, 1, 2, 5, 0)$$

pues

1. Los coeficientes no negativos en la función z a minimizar garantizan que no se puede alcanzar ningún valor por debajo de -7, al ser x_9, x_3 y x_4 no negativas. Este mínimo se obtiene para $x_9 = x_3 = x_4 = 0$.

2. Los valores de las variables básicas, que pueden calcularse usando las restricciones y sustituyendo $x_9 = x_3 = x_4 = 0$, son no negativos; así, también satisfacen la condición de no negatividad. Nótese que esto sucede porque los términos independientes en las restricciones son no negativos.

En consecuencia, si se tiene una función que debe minimizarse con todos sus coeficientes no negativos, entonces, el valor mínimo se consigue cuando todas las variables que intervienen toman el valor cero. Esta solución es factible únicamente si los términos independientes que aparecen en las restricciones son no negativos.

El método simplex empieza con una función a minimizar y un conjunto de restricciones que normalmente no verifican estas condiciones. En la etapa reguladora se transforma el conjunto de restricciones en otro con términos independientes no negativos, y en la conocida como de iteraciones estándar se persigue conseguir coeficientes no negativos en la función transformada que debe optimizarse, mientras que al mismo tiempo se busca conservar los términos independientes no negativos. Si esto es posible, se obtiene la solución óptima. Si no, el problema es no acotado o no factible.

6.2.2 Descripción general

Existen muchas variantes del método simplex, aunque todas ellas se basan en la misma idea central. En este capítulo se describe una de tales versiones.

El método simplex se aplica a un PPL en el formato estándar siguiente. Minimizar

$$f(\mathbf{x}) = \mathbf{c}^T\mathbf{x} \tag{6.3}$$

sujeto a

$$\mathbf{A}\mathbf{x} = \mathbf{b}; \;\; \mathbf{b} \geq \mathbf{0} \tag{6.4}$$

y

$$\mathbf{x} \geq \mathbf{0} \tag{6.5}$$

donde $\mathbf{c} = (c_1, c_2, \ldots, c_n)^T$ es la matriz columna de los coeficientes de la función objetivo, $\mathbf{x} = (x_1, x_2, \ldots, x_n)^T$ es el vector columna de las variables iniciales, y $\mathbf{A}$ es una matriz $m \times n$ que contiene los coeficientes de las restricciones.

Los teoremas 4.1 y 4.2, respectivamente, aseguran que la solución óptima de un PPL, si existe, se alcanza en una solución básica factible (un punto extremo). El MS genera una sucesión ordenada de soluciones básicas factibles que progresivamente mejora el valor de la función objetivo. El MS opera en dos fases:

1. Una etapa de iniciación en la que

(a) El conjunto inicial de restricciones de igualdad se transforma en otro equivalente del mismo tipo (restricciones de igualdad), asociado a una solución básica.

(b) Los valores de las variables básicas se transforman en valores no negativos (se obtiene así una solución básica factible). Este proceso se llamará *fase de regulación*.

2. Una etapa iterativa estándar, en la que los coeficientes de la función objetivo se transforman en valores no negativos y el valor del coste se mejora progresivamente hasta que se obtiene la solución óptima, no se detecta ninguna solución factible, o aparecen soluciones no acotadas. En este proceso iterativo, se calculan distintas soluciones básicas factibles. Para este fin se usa la operación elemental de la pivotación.

6.2.3 Etapa de iniciación

Una peculiaridad esencial del MS consiste en que incorpora una nueva variable Z, idéntica a la función objetivo del problema, y la restricción asociada

$$Z = c_1 x_1 + \cdots + c_n x_n \tag{6.6}$$

Usando la notación típica en el MS, las ecuaciones para las restricciones se expresan como

$$\begin{pmatrix} \mathbf{B} & \mathbf{N} \end{pmatrix} \begin{pmatrix} \mathbf{x_B} \\ - \\ \mathbf{x_N} \end{pmatrix} = \mathbf{b} \tag{6.7}$$

y la función objetivo como

$$Z = \begin{pmatrix} \mathbf{c}_\mathbf{B}^T & \mathbf{c}_\mathbf{N}^T \end{pmatrix} \begin{pmatrix} \mathbf{x_B} \\ - \\ \mathbf{x_N} \end{pmatrix} \tag{6.8}$$

o

$$Z = \begin{pmatrix} 0 & \mathbf{c}_\mathbf{B}^T & \mathbf{c}_\mathbf{N}^T \end{pmatrix} \begin{pmatrix} 1 \\ - \\ \mathbf{x_B} \\ - \\ \mathbf{x_N} \end{pmatrix} \tag{6.9}$$

donde $(\mathbf{B} \quad \mathbf{N})$ es una partición de la matriz $\mathbf{A}$ y $\mathbf{x_B}$ y $\mathbf{x_N}$ definen otra partición de $\mathbf{x}$, en las variables básicas y no básicas, respectivamente.

Usando (6.7), se llega a

$$\mathbf{B}\mathbf{x_B} + \mathbf{N}\mathbf{x_N} = \mathbf{b} \tag{6.10}$$

y

$$\mathbf{x_B} = \mathbf{B}^{-1}\mathbf{b} - \mathbf{B}^{-1}\mathbf{N}\mathbf{x_N} = \mathbf{v} + \mathbf{U}\mathbf{x_N} \tag{6.11}$$

donde

$$\mathbf{v} = \mathbf{B}^{-1}\mathbf{b} \tag{6.12}$$

y

$$\mathbf{U} = -\mathbf{B}^{-1}\mathbf{N} \tag{6.13}$$

Por otro lado, de (6.8) y (6.11), se tiene

$$Z = \mathbf{c}_{\mathbf{B}}^T(\mathbf{v} + \mathbf{U}\mathbf{x}_{\mathbf{N}}) + \mathbf{c}_{\mathbf{N}}^T\mathbf{x}_{\mathbf{N}} \tag{6.14}$$

y

$$Z = \mathbf{c}_{\mathbf{B}}^T\mathbf{v} + (\mathbf{c}_{\mathbf{B}}^T\mathbf{U} + \mathbf{c}_{\mathbf{N}}^T)\mathbf{x}_{\mathbf{N}} \tag{6.15}$$

y

$$Z = u_0 + \mathbf{w}\mathbf{x}_{\mathbf{N}} \tag{6.16}$$

donde

$$u_0 = \mathbf{c}_{\mathbf{B}}^T\mathbf{v} \tag{6.17}$$

y

$$\mathbf{w} = \mathbf{c}_{\mathbf{B}}^T\mathbf{U} + \mathbf{c}_{\mathbf{N}}^T \tag{6.18}$$

Las ecuaciones (6.11) y (6.16) juntas son

$$\left\{ \begin{array}{l} Z = u_0 + \mathbf{w}\mathbf{x}_{\mathbf{N}} \\ \mathbf{x}_{\mathbf{B}} = \mathbf{v} + \mathbf{U}\mathbf{x}_{\mathbf{N}} \end{array} \right. \tag{6.19}$$

y finalmente

$$\begin{pmatrix} Z \\ \mathbf{x}_{\mathbf{B}} \end{pmatrix} = \begin{pmatrix} u_0 & \mathbf{w} \\ \mathbf{v} & \mathbf{U} \end{pmatrix} \begin{pmatrix} 1 \\ \mathbf{x}_{\mathbf{N}} \end{pmatrix} \tag{6.20}$$

El MS comienza con un conjunto de restricciones (6.4) y (6.6) escritas como

$$\begin{pmatrix} Z \\ -- \\ \mathbf{x}_B \end{pmatrix} = \begin{pmatrix} u_0^{(0)} & | & \mathbf{w}^{(0)^T} \\ - & + & - \\ \mathbf{v}^{(0)} & | & \mathbf{U}^{(0)} \end{pmatrix} \begin{pmatrix} 1 \\ -- \\ \mathbf{x}_N \end{pmatrix} = \mathbf{Z}^{(0)} \begin{pmatrix} 1 \\ -- \\ \mathbf{x}_N \end{pmatrix} \tag{6.21}$$

donde $\mathbf{x}_B \cup \mathbf{x}_N$ es una partición del conjunto de variables $\{x_1, x_2, \ldots, x_n\}$, las matrices $\mathbf{v}^{(0)}$ y $\mathbf{U}^{(0)}$ se obtienen resolviendo (6.4) para $\mathbf{x}_B$, y teniendo en cuenta (6.12) y (6.13). Sustituyendo $\mathbf{v}^{(0)}$ y $\mathbf{U}^{(0)}$ en (6.3) se obtiene:

$$u^{(0)} = \mathbf{c}_B^T\mathbf{v}^{(0)}, \quad \mathbf{w}^{(0)} = \mathbf{c}_B^T\mathbf{U}^{(0)} + \mathbf{c}_N^T \tag{6.22}$$

donde $\mathbf{c}_B^T$ y $\mathbf{c}_N^T$ son los coeficientes de la función objetivo asociados a $\mathbf{x}_B$ y $\mathbf{x}_N$, respectivamente.

Ahora, el MS sustituye (6.21), en cada iteración, por un nuevo conjunto equivalente de restricciones con la misma estructura

$$\begin{pmatrix} Z^{(t)} \\ -- \\ \mathbf{x}_B^{(t)} \end{pmatrix} = \begin{pmatrix} u_0^{(t)} & | & \mathbf{w}^{(t)^T} \\ - & + & - \\ \mathbf{v}^{(t)} & | & \mathbf{U}^{(t)} \end{pmatrix} \begin{pmatrix} 1 \\ -- \\ \mathbf{x}_N^{(t)} \end{pmatrix} = \mathbf{Z}^{(t)} \begin{pmatrix} 1 \\ -- \\ \mathbf{x}_N^{(t)} \end{pmatrix} \tag{6.23}$$

donde t indica el número de la iteración y $t = 0$ se refiere a la iteración inicial.

La transformación más importante del MS es la operación elemental de pivotación que se describe a continuación.

6.2.4 Operación elemental de pivotación

El MS llega a la solución del PPL mediante operaciones elementales, que inicialmente actúan sobre $\mathbf{Z}^{(0)}$, para obtener $\mathbf{Z}^{(1)}$ y sucesivamente $\mathbf{Z}^{(2)}, \ldots, \mathbf{Z}^{(t)}$, hasta que se alcanza la solución o se concluye que ésta no existe.

La idea consiste en sustituir una variable básica por una no básica, y el fundamento teórico de este cambio se da en el siguiente teorema.

Teorema 6.1 (propiedad de invarianza de la transformación de pivotación). *Sea $z_{\alpha\beta}^{(t)}$ un elemento no nulo de $\mathbf{Z}^{(t)}$, denominado pivote. Si se transforma el sistema de restricciones $\mathbf{y}_B^{(t)} = \mathbf{Z}^{(t)}\mathbf{y}_N^{(t)}$ en $\mathbf{y}_B^{(t+1)} = \mathbf{Z}^{(t+1)}\mathbf{y}_N^{(t+1)}$, mediante*

$$z_{ij}^{(t+1)} = \left\{ \begin{array}{ll} \dfrac{z_{i\beta}^{(t)}}{z_{\alpha\beta}^{(t)}}, & \text{si} \quad i \neq \alpha, \;\; j = \beta \\ z_{ij}^{(t)} - \dfrac{z_{\alpha j}^{(t)}}{z_{\alpha\beta}^{(t)}} z_{i\beta}^{(t)} & \text{si} \quad i \neq \alpha, \;\; j \neq \beta \\ \dfrac{1}{z_{\alpha\beta}^{(t)}} & \text{si} \quad i = \alpha, \;\; j = \beta \\ -\dfrac{z_{\alpha j}^{(t)}}{z_{\alpha\beta}^{(t)}} & \text{si} \quad i = \alpha, \;\; j \neq \beta \end{array} \right\} \tag{6.24}$$

entonces ambos sistemas de restricciones son equivalentes, esto es, tienen las mismas soluciones (*mismo conjunto factible*). ∎

Demostración. Considérese el conjunto $\mathbf{y}_B^{(t)} = \mathbf{Z}^{(t)}\mathbf{y}_N^{(t)}$. Si se obtiene y_β de la ecuación asociada a la fila α de $\mathbf{Z}^{(t)}$

$$y_\alpha = \mathbf{z}_\alpha^{(t)} \; \mathbf{y}_N^{(t)} = \sum_j z_{\alpha j}^{(t)} y_{N j}$$

se encuentra

$$y_\beta = \sum_{j \neq \beta} \left(\frac{-z_{\alpha j}^{(t)}}{z_{\alpha\beta}^{(t)}} y_{N j} \right) + \frac{1}{z_{\alpha\beta}^{(t)}} y_\alpha = \mathbf{z}_\beta^{(t+1)} \; \mathbf{y}_N^{(t+1)} \tag{6.25}$$

y sustituyendo y_β en el resto de restricciones, se tiene

$$y_i = \sum_{j \neq \beta} \left(z_{ij}^{(t)} - \frac{z_{\alpha j}^{(t)}}{z_{\alpha\beta}^{(t)}} z_{i\beta}^{(t)} \right) y_{N j} + \frac{z_{i\beta}^{(t)}}{z_{\alpha\beta}^{(t)}} y_\alpha = \sum_{j \neq \beta} z_{ij}^{(t+1)} \; y_{N j} + z_{i\alpha}^{(t+1)} \; y_\alpha; \;\; i \neq \beta. \tag{6.26}$$

Así, de (6.25) y (6.26) se llega a

$$\mathbf{y} = \mathbf{Z}^{(t+1)} \; \mathbf{y}_N^{(t+1)}$$

La transformación (6.24), es decir, (6.25) y (6.26), pueden escribirse en forma matricial como

$$\mathbf{Z}^{(t+1)} = \mathbf{Z}^{(t)}\mathbf{T}^{(t)} + \mathbf{S}^{(t)}$$

donde $\mathbf{T}^{(t)}$ es una matriz identidad con su fila β cambiada por

$$\frac{1}{z_{\alpha\beta}} \left(-z_{\alpha 1} \quad \cdots \quad -z_{\alpha\beta-1} \quad 1 \quad -z_{\alpha\beta+1} \quad \cdots \quad -z_{\alpha n} \right) \tag{6.27}$$

y $\mathbf{S}^{(t)}$, una matriz nula con su fila α sustituida por

$$\frac{1}{z_{\alpha\beta}} \left(-z_{\alpha 1} \quad \cdots \quad -z_{\alpha\beta-1} \quad 1 - z_{\alpha\beta} \quad -z_{\alpha\beta+1} \quad \cdots \quad -z_{\alpha n} \right) \tag{6.28}$$

Como todos estos cambios son lineales y tienen matrices asociadas no singulares, conducen a un sistema equivalente de ecuaciones.

La tabla 6.1 muestra las matrices inicial y transformada: los conjuntos de restricciones inicial y transformado.

Obsérvese que, tras esta transformación, los papeles de las variables x_α y x_β se han intercambiado, concretamente, x_α cambia de básica a no básica, y x_β, de no básica a básica.

■

6.2.5 Identificación de una solución óptima

Cuando el PPL se expresa en la forma (6.23) se puede saber fácilmente si se ha encontrado una solución óptima. El siguiente resultado responde a esta cuestión.

Teorema 6.2 (se ha encontrado una solución). *Si*

$$\mathbf{w}^{(t)} \geq \mathbf{0}, \ \ \mathbf{v}^{(t)} \geq \mathbf{0} \tag{6.29}$$

entonces, una solución óptima del PPL es

$$Z^{(t)} = u_0^{(t)}; \ \mathbf{x}_B^* = \mathbf{v}^{(t)}, \ \ \mathbf{x}_N^{(t)} = \mathbf{0} \tag{6.30}$$

donde el asterisco indica solución óptima. ■

Demostración. De (6.23) se tiene

$$Z = u_0^{(t)} + {\mathbf{w}^{(t)}}^T \mathbf{x}_N^{(t)} \tag{6.31}$$

Como todos los elementos en $\mathbf{w}^{(t)}$ son no negativos, el mínimo de la suma del lado derecho de (6.31) es $u_0^{(t)}$ y se alcanza en $\mathbf{x}_N^{(t)} = \mathbf{0}, \mathbf{x}_B^{(t)} = \mathbf{v}^{(t)}$. Como, además, $\mathbf{v}^{(t)} \geq \mathbf{0}$, $\{\mathbf{x}_N^{(t)} = \mathbf{0}; \mathbf{x}_B^{(t)} = \mathbf{v}^{(t)}\}$ es la solución óptima. ■

Esto implica que la solución aparece en la primera columna de la matriz de los coeficientes de las restricciones, y el valor máximo de la función objetivo es $u_0^{(t)}$. Esto es lo que se ha ilustrado con el ejemplo 6.2.1.

Las condiciones (6.29) justifican las dos etapas siguientes:

Tabla 6.1: Matrices inicial y transformada

Matriz inicial					
	x_{N1}	$\dots$	x_β	$\dots$	x_{Nm-n}
x_{B1}	$z_{11}^{(t)}$	$\dots$	$z_{1\beta}^{(t)}$	$\dots$	$z_{1m-n}^{(t)}$
$\vdots$	$\vdots$	$\dots$	$\vdots$	$\dots$	$\vdots$
x_α	$z_{\alpha 1}^{(t)}$	$\dots$	$z_{\alpha\beta}^{(t)}$	$\dots$	$z_{\alpha m-n}^{(t)}$
$\vdots$	$\vdots$	$\dots$	$\vdots$	$\dots$	$\vdots$
x_{Bn}	$z_{n1}^{(t)}$	$\dots$	$z_{n\beta}^{(t)}$	$\dots$	$z_{nm-n}^{(t)}$
Matriz transformada					
	x_{N1}	$\dots$	x_α	$\dots$	x_{Nm-n}
x_{B1}	$z_{11}^{(t)} - \frac{z_{\alpha 1}^{(t)}}{z_{\alpha\beta}^{(t)}} z_{1\beta}^{(t)}$	$\dots$	$\frac{z_{1\beta}^{(t)}}{z_{\alpha\beta}^{(t)}}$	$\dots$	$z_{1m-n}^{(t)} - \frac{z_{\alpha m-n}^{(t)}}{z_{\alpha\beta}^{(t)}} z_{1\beta}^{(t)}$
$\vdots$	$\vdots$	$\dots$	$\vdots$	$\dots$	$\vdots$
x_β	$-\frac{z_{\alpha 1}^{(t)}}{z_{\alpha\beta}^{(t)}}$	$\dots$	$\frac{1}{z_{\alpha\beta}^{(t)}}$	$\dots$	$-\frac{z_{\alpha m-n}^{(t)}}{z_{\alpha\beta}^{(t)}}$
$\vdots$	$\vdots$	$\dots$	$\vdots$	$\dots$	$\vdots$
x_{Bn}	$z_{n1}^{(t)} - \frac{z_{\alpha 1}^{(t)}}{z_{\alpha\beta}^{(t)}} z_{n\beta}^{(t)}$	$\dots$	$\frac{z_{n\beta}^{(t)}}{z_{\alpha\beta}^{(t)}}$	$\dots$	$z_{nm-n}^{(t)} - \frac{z_{\alpha m-n}^{(t)}}{z_{\alpha\beta}^{(t)}} z_{n\beta}^{(t)}$

1. **Iteración reguladora**. Se usa una iteración reguladora para conseguir $\mathbf{v}^{(t)} \geq \mathbf{0}$; es decir, se buscan valores no negativos para las variables básicas.

2. **Iteraciones estándar**. Se realizan transformaciones elementales tales que

   (a) Se verifica la condición
   $$\mathbf{v}^{(t)} \geq \mathbf{0} \tag{6.32}$$
   y

   (b) mejora el valor de la función objetivo al exigir la condición
   $$\mathbf{w}^{(t)} \geq \mathbf{0} \tag{6.33}$$
   lo que implica una solución acotada, o se detecta la no acotación o la carencia de solución óptima.

Esta transformación consiste en cambiar la partición de $\mathbf{x}$ en variables básicas y no básicas de modo que una de las variables básicas y una de las no básicas se intercambian entre sí.

6.2.6 Iteración reguladora

En la iteración reguladora, el sistema inicial de restricciones se convierte en otro que verifica

$$\mathbf{v}^{(t)} \geq \mathbf{0}$$

esto es, se obtiene una solución básica factible.

Se pueden dar dos posibilidades:

1. Existe una variable no básica x_β, tal que se cumple una de las siguientes condiciones

$$\left\{ \begin{array}{lll} u_{i\beta}^{(0)} > 0 & \text{si} & v_i^{(0)} < 0 \\ u_{i\beta}^{(0)} \geq 0 & \text{si} & v_i^{(0)} \geq 0 \end{array} \right. \quad \forall i \tag{6.34}$$

2. No existe tal variable. En este caso se introduce una variable adicional $x_{n+1} \geq 0$, se suma el término Mx_{n+1} a la función objetivo, donde M es una constante positiva suficientemente grande, y se modifican las restricciones añadiendo el término x_{n+1} a todas las variables básicas. Esto supone añadir un nuevo elemento, M, a la matriz $\mathbf{w}$, y añadir una columna de unos a la matriz $\mathbf{U}^{(0)}$, que satisface (6.34), y elegir $x_\beta = x_{n+1}$.

En ambos casos, se verifica (6.34) después de usar la transformación de pivotación (6.24) con el pivote $u_{\alpha\beta}^{(0)}$ dado por

$$\frac{v_\alpha^{(0)}}{u_{\alpha\beta}^{(0)}} = \min_{v_i^{(0)}<0} \frac{v_i^{(0)}}{u_{i\beta}^{(0)}} \tag{6.35}$$

Nota 6.1 *En la situación 2 anterior, y con una elección apropiada de M, si el PPL es factible, entonces $x_{n+1} = 0$.* ■

6.2.7 Detección de no acotación

En este apartado se explica cómo se puede concluir que el problema es no acotado.

Sea, por ejemplo, el PPL siguiente. Minimizar

$$Z = -x_2$$

sujeto a

$$\begin{array}{lcrrr} x_3 & = & & x_1 & \\ x_4 & = & & & x_2 \\ x_5 & = & -1 & +x_1 & +x_2 \end{array}$$

Nótese que este problema es no acotado pues se puede hacer que el valor de x_2 sea tan grande como se quiera, manteniendo las demás variables no básicas en un valor nulo, para obtener valores no negativos de las variables básicas y valores no acotados de la función objetivo.

Lo anterior sucede cuando existe una columna β con todos sus valores no negativos y $w_\beta < 0$:

$$w_\beta < 0 \text{ y } u_{i\beta} \geq 0 \; \forall i$$

En el caso general y en la iteración t se obtiene un valor de la función objetivo

$$Z = u_0^{(t)} + \mathbf{w}^{(t)^T} \mathbf{x}_N^{(t)} \tag{6.36}$$

que puede hacerse tan pequeño como se desee, pero manteniendo la solución en la región factible, pues

$$\mathbf{x}_B^{(t)} = \mathbf{v}^{(t)} + \mathbf{U}^{(t)} \mathbf{x}_N^{(t)} \geq \mathbf{0}$$

6.2.8 Detección de no factibilidad

En esta subsección se explica cómo puede detectarse la no factibilidad.

Esto sucede cuando la condición de solución se da con $x_{n+1} > 0$ (véase el comentario 6.1).

6.2.9 Etapa de iteraciones estándar

Con las iteraciones estándar del MS, la función objetivo se transforma en una función con todos sus coeficientes no negativos; es decir, se convierte $\mathbf{w}^{(0)}$ en $\mathbf{w}^{(1)} \geq \mathbf{0}$.

En esta fase, se usa la transformación (6.24) para obtener $\mathbf{Z}^{(2)}, \ldots, \mathbf{Z}^{(t)}$, de modo que:

1. Se consigue un sistema equivalente de restricciones:

$$\mathbf{x}_B^{(0)} = \mathbf{v}^{(0)} + \mathbf{U}^{(0)} \, \mathbf{x}_N^{(0)} \Leftrightarrow \mathbf{x}_B^{(t)} = \mathbf{v}^{(t)} + \mathbf{U}^{(t)} \, \mathbf{x}_N, \forall t \geq 1 \tag{6.37}$$

2. Se sigue verificando la condición

$$\mathbf{v}^{(t)} \geq \mathbf{0} \tag{6.38}$$

3. El valor de la función objetivo no aumenta:

$$z^{(2)} \geq \cdots \geq z^{(t)} \tag{6.39}$$

4. Los pivotes se seleccionan para maximizar la disminución del valor de la función objetivo.

De esta manera, cada iteración del PPL es equivalente a la minimización de Z sujeto a

$$\mathbf{x}_B = \mathbf{v}^{(t)} + \mathbf{U}^{(t)} \, \mathbf{x}_N; \; \mathbf{x}_N \geq \mathbf{0}, \mathbf{x}_B \geq \mathbf{0} \tag{6.40}$$

En la etapa de iteraciones estándar el MS se desarrolla del siguiente modo:

1. Cada iteración tiene asociado un punto factible, que se obtiene asignando el valor cero al conjunto de variables no básicas, $\mathbf{x}_N$. Como $\mathbf{x}_N^{(t)} = \mathbf{0}$, $u_0^{(t)}$ siempre contiene el valor de la función objetivo, y $\mathbf{x}_B$, los valores de las variables básicas.

2. El fundamento del MS consiste en cambiar de manera adecuada los conjuntos de variables básicas y no básicas en los distintos pasos del método. En cada iteración, una variable básica, x_α, y una variable no básica, x_β, intercambian sus papeles. Este cambio se realiza de modo que el nuevo conjunto de restricciones sea equivalente al anterior, en el sentido de que tengan las mismas soluciones. Por tanto se necesita

   (a) Concretar una regla para elegir la variable x_α que entra en el conjunto $\mathbf{x}_N$ y abandona el conjunto $\mathbf{x}_B$.

   (b) Dar una regla para elegir la variable x_β que sale de $\mathbf{x}_N$ y entra en $\mathbf{x}_B$.

3. Esta selección de las variables entrante y saliente se lleva a cabo disminuyendo el valor de la función objetivo, a la vez que se busca un punto factible.

Una vez que se han seleccionado las variables entrante y saliente, se transforma el sistema de restricciones en un sistema equivalente, usando la transformación de pivotación.

Selección de la variable x_β para formar parte de $\mathbf{x}_B$

De acuerdo con nuestros resultados previos, si la solución no se ha alcanzado en el estado t, y la no factibilidad o falta de acotación no se han detectado aún, deben existir unos ciertos índices i y j tales que $w_j^{(t)} < 0$ y $u_{ij}^{(t)} < 0$.

La variable x_β que sale de $\mathbf{x}_N$ es la que verifica

$$w_\beta^{(t)} = \min_{w_j^{(t)}<0} w_j^{(t)} \tag{6.41}$$

Nótese que, según se realiza la transformación de pivotación, el nuevo valor de $v_\alpha^{(t)}$ es

$$v_\alpha^{(t+1)} = -\frac{v_\alpha^{(t)}}{u_{\alpha\beta}} \geq 0 \tag{6.42}$$

De esta manera, como se ha establecido en nuestras hipótesis anteriores, se tiene que

$$z^{(0)} \geq \cdots \geq z^{(t)} \Rightarrow f(\mathbf{x}^{(0)}) \geq \cdots \geq f(\mathbf{x}^{(t)})$$

una relación que, interpretada en un sentido geométrico, proporciona una sucesión de puntos $\mathbf{x}^{(0)}, \ldots, \mathbf{x}^{(t)}$, que acercan los hiperplanos $f(\mathbf{x}) = f(\mathbf{x}^{(t)})$ al punto óptimo $\mathbf{x}^* \in X$.

Selección de la variable x_α que abandona $\mathbf{x}_B$

Una vez conocida la variable entrante x_β, se selecciona el pivote $u_{\alpha\beta}^{(t)}$ de la transformación (6.24) para que $\mathbf{v}^{(t)}$ satisfaga (6.38). Así, la fila del pivote α y el valor del pivote $u_{\alpha\beta}^{(t)}$ se obtienen mediante

$$\Lambda_\alpha^{(t)} = \frac{v_\alpha^{(t)}}{u_{\alpha\beta}^{(t)}} = \max_{u_{i\beta}^{(t)}<0} \frac{v_i^{(t)}}{u_{i\beta}^{(t)}} \leq 0 \tag{6.43}$$

donde $\Lambda_\alpha^{(t)}$ es la *razón función-pivote*. Lo anterior garantiza que $\mathbf{v}^{t+1} \geq 0$. Por lo tanto, la variable asociada a la fila α abandona el conjunto $\mathbf{x}_B$.

A continuación se justifica (6.43). Según la transformación de pivotación (6.24), las variables nuevas $v_i^{(t+1)}$ deben ser no negativas para todo i, es decir, se debe cumplir

$$v_\alpha^{(t+1)} = -\frac{v_\alpha^{(t)}}{u_{\alpha\beta}^{(t)}} \geq 0 \tag{6.44}$$

$$v_i^{(t+1)} = v_i^{(t)} - \Lambda_\alpha^{(t)} u_{i\beta}^{(t)} \geq 0; \ \ \forall i \neq \alpha \tag{6.45}$$

La condición (6.44) es correcta debido a (6.43). Para comprobar que (6.45) también es cierto, se tienen dos posibilidades:

1. Si $u_{i\beta}^{(t)} > 0$, entonces se cumple (6.45), pues $v_i^{(t)}$, $-\Lambda_\alpha^{(t)}$ y $u_{i\beta}^{(t)}$ son no negativos.

2. Si $u_{i\beta}^{(t)} < 0$, debido a la definición de máximo, se tiene

$$\Lambda_\alpha^{(t)} \geq \frac{v_i^{(t)}}{u_{i\beta}^{(t)}} \Leftrightarrow u_{i\beta}^{(t)} \Lambda_\alpha^{(t)} \leq v_i^{(t)} \tag{6.46}$$

   lo que implica (6.45).

6.2.10 El algoritmo simplex revisado

Toda la argumentación precedente nos permite construir nuestra versión particular del algoritmo simplex revisado, que trabaja con la matriz $\mathbf{U}^{(t)}$, hasta obtener la solución óptima o detectar la falta de puntos factibles o la no acotación.

Algoritmo 6.1 (el algoritmo simplex)

- **Entrada.** La tabla inicial, $\{\mathbf{U}^{(0)}, \mathbf{v}^{(0)}, \mathbf{w}^{(0)}, u^{(0)}\}$ del PPL, conteniendo la función objetivo, y las restricciones del problema.

- **Salida.** La solución del problema de minimización, o un mensaje advirtiendo sobre la no admisibilidad o la falta de acotación.

Iteración reguladora

Paso 1 (iniciación). Si $\mathbf{v}^{(0)} \geq \mathbf{0}$, ir al paso 4. En otro caso, transformar $\mathbf{v}^{(0)}$ para que todos sus elementos sean no negativos. Con este fin, comprobar si existe una columna β en $\mathbf{U}^{(0)}$ que verifique una de las dos condiciones

$$\left\{\begin{array}{lll} u_{i\beta}^{(0)} > 0 & \text{si} & v_i^{(0)} < 0 \\ u_{i\beta}^{(0)} \geq 0 & \text{si} & v_i^{(0)} \geq 0 \end{array}\right. \quad \forall i \tag{6.47}$$

Si la respuesta es afirmativa, continuar en el paso 2. En otro caso, introducir la variable artificial $x_{n+1} \geq 0$, modificar el valor de la función objetivo, sumando el término Mx_{n+1}, donde M es una constante positiva grande, sumar el término x_{n+1} a todas las restricciones (variables básicas), y elegir $x_\beta = x_{n+1}$.
Paso 2 (búsqueda del pivote). Encontrar la fila del pivote α usando

$$\frac{v_\alpha^{(0)}}{u_{\alpha\beta}^{(0)}} = \min_{v_i^{(0)} < 0} \frac{v_i^{(0)}}{u_{i\beta}^{(0)}} \tag{6.48}$$

e ir al paso 3.

Iteraciones estándar

Paso 3 (pivotación). Realizar la transformación de pivotación

$$z_{ij}^{(t+1)} = \left\{\begin{array}{ll} \dfrac{z_{i\beta}^{(t)}}{z_{\alpha\beta}^{(t)}}, & \text{si} \quad i \neq \alpha, \ \ j = \beta \\ z_{ij}^{(t)} - \dfrac{z_{\alpha j}^{(t)}}{z_{\alpha\beta}^{(t)}} z_{i\beta}^{(t)} & \text{si} \quad i \neq \alpha, \ \ j \neq \beta \\ \dfrac{1}{z_{\alpha\beta}^{(t)}} & \text{si} \quad i = \alpha, \ \ j = \beta \\ -\dfrac{z_{\alpha j}^{(t)}}{z_{\alpha\beta}^{(t)}} & \text{si} \quad i = \alpha, \ \ j \neq \beta \end{array}\right\} \tag{6.49}$$

Paso 4 (selección de la variable entrante x_β). Si la condición de solución

$$\mathbf{w}^{(t)} \geq \mathbf{0}, \ \ \mathbf{v}^{(t)} \geq \mathbf{0} \tag{6.50}$$

es cierta y se verifica además la condición de no acotación, $x_{n+1} > 0$, el problema es no factible; si la condición de solución se cumple con $x_{n+1} = 0$, la solución es

$$\mathbf{x}_B^* = \mathbf{v}^{(t)}; \ \mathbf{x}_N^* = \mathbf{0} \tag{6.51}$$

y el algoritmo concluye. En otro caso, se selecciona la variable entrante x_β, formando parte de $\mathbf{x}_B$, mediante

$$w_\beta^{(t)} = \min_{w_j^{(t)}<0} w_j^{(t)} \tag{6.52}$$

Paso 5 (selección de la variable saliente). Si todo $u_{i\beta} \geq 0$, el problema es no acotado. En otro caso, se selecciona la variable saliente x_α que deja $\mathbf{x}_B$ utilizando

$$\frac{v_\alpha^{(t)}}{u_{\alpha\beta}^{(t)}} = \max_{u_{i\beta}^{(t)}<0} \frac{v_i^{(t)}}{u_{i\beta}^{(t)}} \tag{6.53}$$

Volver al paso 3. ■

Nota 6.2 *Si el conjunto de restricciones se dan como desigualdades, la tabla inicial puede escribirse directamente a partir de la función objetivo y los coeficientes de la matriz de desigualdades, según se muestra en el ejemplo siguiente.* ■

6.2.11 Algunos ejemplos ilustrativos

En esta sección se ilustra el MS mediante varios ejemplos.

Ejemplo 6.1 (primer ejemplo). Examínese de nuevo el PPL siguiente. Minimizar

$$Z = -3x_1 - x_2 \tag{6.54}$$

sujeto a

$$\begin{array}{rrcr} -3x_1 & +2x_2 & \leq & 2 \\ x_1 & +x_2 & \leq & 6 \\ x_1 & & \leq & 3 \\ 2x_1 & -x_2 & \leq & 4 \\ & -x_2 & \leq & 0 \\ -x_1 & -x_2 & \leq & -1 \end{array} \tag{6.55}$$

con

$$x_1, x_2 \geq 0. \tag{6.56}$$

En primer lugar es necesario transformarlo en un conjunto de igualdades usando para ello seis variables de holgura

$$\begin{array}{rrrrrrrrcr} -3x_1 & +2x_2 & +x_3 & & & & & & = & 2 \\ x_1 & +x_2 & & +x_4 & & & & & = & 6 \\ x_1 & & & & +x_5 & & & & = & 3 \\ 2x_1 & -x_2 & & & & +x_6 & & & = & 4 \\ & -x_2 & & & & & +x_7 & & = & 0 \\ -x_1 & -x_2 & & & & & & +x_8 & = & -1 \end{array} \tag{6.57}$$

Tabla 6.2: Tabla inicial con las matrices $u^{(0)}, \mathbf{w}^{(0)}, \mathbf{v}^{(0)}$, y $\mathbf{U}^{(0)}$ y tabla transformada después de añadir la variable artificial x_9 (Ejemplo 6.1)

$\mathbf{Z}^{(0)}$				$\mathbf{Z}^{(1)}$				
	1	x_1	x_2		1	x_1	x_2	x_9
Z	0	–3	–1	Z	0	–3	–1	20
x_3	2	3	–2	x_3	2	3	–2	1
x_4	6	–1	–1	x_4	6	–1	–1	1
x_5	3	–1	0	x_5	3	–1	0	1
x_6	4	–2	1	x_6	4	–2	1	1
x_7	0	0	1	x_7	0	0	1	1
x_8	–1	1	1	x_8	–1	1	1	**1**

con

$$x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8 \geq 0 \tag{6.58}$$

Las restricciones (6.57) pueden expresarse como

$$\begin{array}{rcrrr} x_3 & = & 2 & +3x_1 & -2x_2 \\ x_4 & = & 6 & -x_1 & -x_2 \\ x_5 & = & 3 & -x_1 & \\ x_6 & = & 4 & -2x_1 & +x_2 \\ x_7 & = & & & +x_2 \\ x_8 & = & -1 & +x_1 & +x_2 \end{array} \tag{6.59}$$

Las expresiones (6.54) y (6.59) conducen al $\mathbf{Z}^{(0)}$ inicial en la tabla 6.2, donde los pivotes aparecen en negrita.

Usando el algoritmo del MS se obtienen los siguientes resultados:

Paso 1. La tabla 6.2 muestra los valores iniciales de $\mathbf{Z}^{(0)}$ y de $\mathbf{u}^{(0)}, \mathbf{w}^{(0)}, \mathbf{v}^{(0)}$, y $\mathbf{U}^{(0)}$, donde se puede ver que algunos de los elementos en la primera columna de $\mathbf{Z}^{(0)}$ son negativos. Por tanto, se necesita transformarlos a valores no negativos. Como ninguna columna β verifica

$$\left\{ \begin{array}{lll} u_{i\beta}^{(0)} > 0 & \text{si} & v_i^{(0)} < 0 \\ u_{i\beta}^{(0)} \geq 0 & \text{si} & v_i^{(0)} \geq 0 \end{array} \right. \quad \forall i$$

se introduce la variable artificial $x_9 \geq 0$, se modifica la función objetivo añadiendo el término $20x_9$, se suma el término x_9 a todas las restricciones (variables básicas), y se hace $x_\beta = x_9$, para obtener $\mathbf{Z}^{(1)}$.

Paso 2. Se busca el pivote usando

$$\frac{v_\alpha^{(0)}}{u_{\alpha\beta}^{(0)}} = \min_{v_i^{(0)}<0} \frac{v_i^{(0)}}{u_{i\beta}^{(0)}} = \min\{\frac{-1}{1}\} = -1 \tag{6.60}$$

lo que supone que $x_\alpha = x_8$.

Paso 3. Se realiza el proceso de pivotación para obtener $\mathbf{Z}^{(2)}$ que se muestra en la tabla 6.3, donde los pivotes se han marcado en negrita. Obsérvese que ahora todos los coeficientes $\mathbf{v}$, en la primera columna, son no negativos (y permanecerán no negativos a partir de ahora).

Paso 4. Como no se satisface aún la condición de terminación, pues no se cumple $\mathbf{w} \geq \mathbf{0}$, se busca la variable entrante usando

$$w_\beta^{(t)} = \min_{w_j^{(t)}<0} w_j^{(t)} = \min\{-23, -21\} = -23 \tag{6.61}$$

lo que implica $x_\beta = x_1$.

Paso 5. Se selecciona la variable saliente mediante

$$\frac{v_\alpha^{(t)}}{u_{\alpha\beta}^{(t)}} = \max_{u_{i\beta}^{(t)}<0} \frac{v_i^{(t)}}{u_{i\beta}^{(t)}} = \max\left\{\frac{7}{-2}, \frac{4}{-2}, \frac{5}{-3}, \frac{1}{-1}, \frac{1}{-1}\right\} = -\frac{1}{1} \tag{6.62}$$

lo que conduce a una relación entre x_7 y x_9, que se resuelve eligiendo arbitrariamente la primera de ellas.

Paso 3. Se realiza el proceso de pivotación y se obtiene $\mathbf{Z}^{(3)}$, que se muestra en la tabla 6.3.

Paso 4. Como, al no ser $\mathbf{w} \geq \mathbf{0}$, no se verifica la condición de solución se procede a encontrar la variable entrante usando

$$w_\beta^{(t)} = \min_{w_j^{(t)}<0} w_j^{(t)} = \min\{-21, -3\} = -21 \tag{6.63}$$

lo que lleva a $x_\beta = x_2$.

Paso 5. Se selecciona la variable saliente mediante

$$\frac{v_\alpha^{(t)}}{u_{\alpha\beta}^{(t)}} = \max_{u_{i\beta}^{(t)}<0} \frac{v_i^{(t)}}{u_{i\beta}^{(t)}} = \max\left\{\frac{5}{-3}, \frac{5}{-2}, \frac{2}{-1}, \frac{0}{-1}\right\} = 0 \tag{6.64}$$

lo que supone $x_\alpha = x_9$.

Paso 3. Se realiza el proceso de pivotación para obtener $\mathbf{Z}^{(4)}$, que se muestra en la tabla 6.3.

Paso 4. Como no se verifica la condición de solución, pues no se cumple $\mathbf{w} \geq \mathbf{0}$, se busca la variable entrante usando

$$w_\beta^{(t)} = \min_{w_j^{(t)}<0} w_j^{(t)} = \min\{-3\} = -3 \tag{6.65}$$

Tabla 6.3: Tablas asociadas a las iteraciones 2 a 7, mostrando $u^{(0)}, \mathbf{w}^{(0)}, \mathbf{v}^{(0)}$, y $\mathbf{U}^{(0)}$ (Ejemplo 6.1). Los pivotes aparecen en negrita.

$\mathbf{Z}^{(2)}$					$\mathbf{Z}^{(3)}$				
	1	x_1	x_2	x_8		1	x_7	x_2	x_8
Z	20	-23	-21	20	Z	-3	23	-21	-3
x_3	3	2	-3	1	x_3	5	-2	-3	3
x_4	7	-2	-2	1	x_4	5	2	-2	-1
x_5	4	-2	-1	1	x_5	2	2	-1	-1
x_6	5	-3	0	1	x_6	2	3	0	-2
x_7	1	**-1**	0	1	x_1	1	-1	0	1
x_9	1	-1	-1	1	x_9	0	1	**-1**	0
$\mathbf{Z}^{(4)}$					$\mathbf{Z}^{(5)}$				
	1	x_7	x_9	x_8		1	x_7	x_9	x_6
Z	-3	2	21	-3	Z	-6	-5/2	21	3/2
x_3	5	-5	3	3	x_3	8	-1/2	3	-3/2
x_4	5	0	2	-1	x_4	4	-3/2	2	1/2
x_5	2	1	1	-1	x_5	1	**-1/2**	1	1/2
x_6	2	3	0	**-2**	x_8	1	3/2	0	-1/2
x_1	1	-1	0	1	x_1	2	1/2	0	-1/2
x_2	0	1	-1	0	x_2	0	1	-1	0
$\mathbf{Z}^{(6)}$					$\mathbf{Z}^{(7)}$				
	1	x_5	x_9	x_6		1	x_5	x_9	x_4
Z	-11	5	16	-1	Z	-12	2	17	1
x_3	7	1	2	-2	x_3	5	-5	4	2
x_4	1	3	-1	**-1**	x_6	1	3	-1	-1
x_7	2	-2	2	1	x_7	3	1	1	-1
x_8	4	-3	3	1	x_8	5	0	2	-1
x_1	3	-1	1	0	x_1	3	-1	1	0
x_2	2	-2	1	1	x_2	3	1	0	-1

lo que implica $x_\beta = x_8$.

Paso 5. Se elige la variable saliente usando

$$\frac{v_\alpha^{(t)}}{u_{\alpha\beta}^{(t)}} = \max_{u_{i\beta}^{(t)}<0} \frac{v_i^{(t)}}{u_{i\beta}^{(t)}} = \max\left\{\frac{5}{-1}, \frac{2}{-1}, \frac{2}{-2}\right\} = -\frac{2}{2} \tag{6.66}$$

lo que lleva a $x_\alpha = x_6$.

Paso 3. El proceso de pivotación nos conduce a $\mathbf{Z}^{(5)}$, que se muestra en la tabla 6.3.

Paso 4. Como no se satisface la condición de solución, pues no se cumple $\mathbf{w} \geq \mathbf{0}$, se procede a buscar la variable entrante usando

$$w_\beta^{(t)} = \min_{w_j^{(t)}<0} w_j^{(t)} = \min\left\{\frac{-5}{2}\right\} = -\frac{5}{2} \tag{6.67}$$

lo que supone $x_\beta = x_7$.

Paso 5. Se selecciona la variable saliente mediante

$$\frac{v_\alpha^{(t)}}{u_{\alpha\beta}^{(t)}} = \max_{u_{i\beta}^{(t)}<0} \frac{v_i^{(t)}}{u_{i\beta}^{(t)}} = \max\left\{\frac{8}{-1/2}, \frac{4}{-3/2}, \frac{1}{-1/2}\right\} = -2 \tag{6.68}$$

lo que supone $x_\alpha = x_5$.

Paso 3. El proceso de pivotación nos lleva a $\mathbf{Z}^{(6)}$, que se muestra en la tabla 6.3.

Paso 4. Como no se satisface la condición de solución, porque no se cumple $\mathbf{w} \geq \mathbf{0}$, se busca la variable entrante usando

$$w_\beta^{(t)} = \min_{w_j^{(t)}<0} w_j^{(t)} = \min\{-1\} = -1 \tag{6.69}$$

lo que lleva a $x_\beta = x_6$.

Paso 5. Se elige la variable saliente usando

$$\frac{v_\alpha^{(t)}}{u_{\alpha\beta}^{(t)}} = \max_{u_{i\beta}^{(t)}<0} \frac{v_i^{(t)}}{u_{i\beta}^{(t)}} = \max\left\{\frac{7}{-2}, \frac{1}{-1}\right\} = -1 \tag{6.70}$$

lo que lleva a $x_\alpha = x_4$.

Paso 3. Se realiza un nuevo paso de pivotación para obtener $\mathbf{Z}^{(7)}$, que se muestra en la tabla 6.3.

Tabla 6.4: Tabla inicial con matrices $u^{(0)}, \mathbf{w}^{(0)}, \mathbf{v}^{(0)}$, y $\mathbf{U}^{(0)}$ (Ejemplo 6.2)

	$\mathbf{Z}^{(0)}$								
	1	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8
Z	0	–2	–4	–3	1	–2	–1	–2	–2
x_9	17	–5	–4	–3	–2	–1	–2	–3	–4
x_{10}	11	–2	1	–4	2	–3	–3	–1	1
x_{11}	8	–3	–2	–3	2	–1	1	–2	1
x_{12}	4	–1	1	–2	–2	1	0	–1	–1
x_{13}	14	–4	–3	–1	1	2	–1	–4	–3

Paso 4. Como la condición de solución se cumple, al ser $\mathbf{w} \geq \mathbf{0}$, y $x_9 = 0$, se ha encontrado la siguiente solución óptima del problema:

$$\begin{array}{rcl} Z & = & -12; \\ (x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9) & = & (3, 3, 5, 0, 0, 1, 3, 5, 0) \end{array}$$

■

Ejemplo 6.2 (un ejemplo más complicado). Considérese el siguiente PPL. Minimizar

$$Z = -2x_1 - 4x_2 - 3x_3 + x_4 - 2x_5 - x_6 - 2x_7 - 2x_8 \tag{6.71}$$

sujeto a

$$\begin{array}{rrrrrrrrcr} 5x_1 & +4x_2 & +3x_3 & +2x_4 & +x_5 & +2x_6 & +3x_7 & +4x_8 & \leq & 17 \\ 2x_1 & -x_2 & +4x_3 & -2x_4 & +3x_5 & +3x_6 & +x_7 & -x_8 & \leq & 11 \\ 3x_1 & +2x_2 & +3x_3 & -2x_4 & +x_5 & -x_6 & +2x_7 & -x_8 & \leq & 8 \\ x_1 & -x_2 & +2x_3 & +2x_4 & -x_5 & & +x_7 & +x_8 & \leq & 4 \\ 4x_1 & +3x_2 & +x_3 & -x_4 & -2x_5 & +x_6 & +4x_7 & +3x_8 & \leq & 14 \end{array} \tag{6.72}$$

con

$$x_1, \ldots, x_8 \geq 0 \tag{6.73}$$

Las tablas 6.4 y 6.5 (en que las entradas de los pivotes se han escrito en negrita), dan las matrices correspondientes a todas las iteraciones. Se deja al lector la aplicación detallada del algoritmo MS y la comprobación de que la solución final es:

$$\begin{array}{rcl} Z & = & \dfrac{-772}{39} \\ (x_1, x_2, \ldots, x_{13}) & = & \left(0, \dfrac{83}{39}, 0, 0, \dfrac{61}{13}, 0, 0, \dfrac{37}{39}, 0, 0, 0, \dfrac{385}{39}, \dfrac{184}{13}\right) \end{array}$$

■

Tabla 6.5: Tablas asociadas a las iteraciones 1 a 4 mostrando $u^{(0)}, \mathbf{w}^{(0)}, \mathbf{v}^{(0)}$, y $\mathbf{U}^{(0)}$ (Ejemplo 6.2)

					$\mathbf{Z}^{(1)}$				
	1	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8
Z	0	–2	–4	–3	1	–2	–1	–2	–2
x_9	17	–5	–4	–3	–2	–1	–2	–3	–4
x_{10}	11	–2	1	–4	2	–3	–3	–1	1
x_{11}	8	–3	**–2**	–3	2	–1	1	–2	1
x_{12}	4	–1	1	–2	–2	1	0	–1	–1
x_{13}	14	–4	–3	–1	1	2	–1	–4	–3
					$\mathbf{Z}^{(2)}$				
	1	x_1	x_{11}	x_3	x_4	x_5	x_6	x_7	x_8
Z	–16	4	2	3	–3	0	–3	2	–4
x_9	1	1	2	3	–6	1	–4	1	**–6**
x_{10}	15	$\frac{-7}{2}$	$\frac{-1}{2}$	$\frac{-11}{2}$	3	$\frac{-7}{2}$	$\frac{-5}{2}$	-2	$\frac{3}{2}$
x_2	4	$\frac{-3}{2}$	$\frac{-1}{2}$	$\frac{-3}{2}$	1	$\frac{-1}{2}$	$\frac{1}{2}$	–1	$\frac{1}{2}$
x_{12}	8	$\frac{-5}{2}$	$\frac{-1}{2}$	$\frac{-7}{2}$	–1	$\frac{1}{2}$	$\frac{1}{2}$	–2	$\frac{-1}{2}$
x_{13}	2	$\frac{1}{2}$	$\frac{3}{2}$	$\frac{7}{2}$	–2	$\frac{7}{2}$	$\frac{-5}{2}$	–1	$\frac{-9}{2}$
					$\mathbf{Z}^{(3)}$				
	1	x_1	x_{11}	x_3	x_4	x_5	x_6	x_7	x_9
Z	$\frac{-50}{3}$	$\frac{10}{3}$	$\frac{2}{3}$	1	1	$\frac{-2}{3}$	$\frac{-1}{3}$	$\frac{4}{3}$	$\frac{2}{3}$
x_8	$\frac{1}{6}$	$\frac{1}{6}$	$\frac{1}{3}$	$\frac{1}{2}$	–1	$\frac{1}{6}$	$\frac{-2}{3}$	$\frac{1}{6}$	$\frac{-1}{6}$
x_{10}	$\frac{61}{4}$	$\frac{-13}{4}$	0	$\frac{-19}{4}$	$\frac{3}{2}$	$\mathbf{\frac{-13}{4}}$	$\frac{-7}{2}$	$\frac{-7}{4}$	$\frac{-1}{4}$
x_2	$\frac{49}{12}$	$\frac{-17}{12}$	$\frac{-1}{3}$	$\frac{-5}{4}$	$\frac{1}{2}$	$\frac{-5}{12}$	$\frac{1}{6}$	$\frac{-11}{12}$	$\frac{-1}{12}$
x_{12}	$\frac{95}{12}$	$\frac{-31}{12}$	$\frac{-2}{3}$	$\frac{-15}{4}$	$\frac{-1}{2}$	$\frac{5}{12}$	$\frac{5}{6}$	$\frac{-25}{12}$	$\frac{1}{12}$
x_{13}	$\frac{5}{4}$	$\frac{-1}{4}$	0	$\frac{5}{4}$	$\frac{5}{2}$	$\frac{11}{4}$	$\frac{1}{2}$	$\frac{-7}{4}$	$\frac{3}{4}$
					$\mathbf{Z}^{(4)}$				
	1	x_1	x_{11}	x_3	x_4	x_{10}	x_6	x_7	x_9
Z	$\frac{-772}{39}$	4	$\frac{2}{3}$	$\frac{77}{39}$	$\frac{9}{13}$	$\frac{8}{39}$	$\frac{5}{13}$	$\frac{22}{13}$	$\frac{28}{39}$
x_8	$\frac{37}{39}$	0	$\frac{1}{3}$	$\frac{10}{39}$	$\frac{-12}{13}$	$\frac{-2}{39}$	$\frac{-11}{13}$	$\frac{1}{13}$	$\frac{-7}{39}$
x_5	$\frac{61}{13}$	–1	0	$\frac{-19}{13}$	$\frac{6}{13}$	$\frac{-4}{13}$	$\frac{-14}{13}$	$\frac{-7}{13}$	$\frac{-1}{13}$
x_2	$\frac{83}{39}$	–1	$\frac{-1}{3}$	$\frac{-25}{39}$	$\frac{4}{13}$	$\frac{5}{39}$	$\frac{8}{13}$	$\frac{-9}{13}$	$\frac{-2}{39}$
x_{12}	$\frac{385}{39}$	–3	$\frac{-2}{3}$	$\frac{-170}{39}$	$\frac{-4}{13}$	$\frac{-5}{39}$	$\frac{5}{13}$	$\frac{-30}{13}$	$\frac{2}{39}$
x_{13}	$\frac{184}{13}$	–3	0	$\frac{-36}{13}$	$\frac{49}{13}$	$\frac{-11}{13}$	$\frac{-32}{13}$	$\frac{-42}{13}$	$\frac{7}{13}$

Ejemplo 6.3 (ejemplo no acotado). Se examina de nuevo el PPL: Minimizar

$$Z = -3x_1 - x_2 \tag{6.74}$$

sujeto a

$$\begin{array}{rrcr} -3x_1 & +2x_2 & \leq & 2 \\ & -x_2 & \leq & 0 \\ -x_1 & -x_2 & \leq & -1 \end{array} \tag{6.75}$$

con

$$x_1, x_2 \geq 0 \tag{6.76}$$

Paso 1. La tabla 6.6 (donde los pivotes aparecen en negrita), muestra los valores iniciales de $\mathbf{Z}^{(0)}$ con las matrices $u^{(0)}, \mathbf{w}^{(0)}, \mathbf{v}^{(0)}$, y $\mathbf{U}^{(0)}$, donde se puede ver que algunos de los elementos en la primera columna son negativos. Por tanto, se necesita transformarlos en valores no negativos.

Como la columna asociada con $x_\beta = x_1$ verifica

$$\left\{ \begin{array}{lll} u_{i\beta} > 0 & \text{si} & v_i < 0 \\ u_{i\beta} \geq 0 & \text{si} & v_i \geq 0 \end{array} \right. \forall i \tag{6.77}$$

se sigue con el paso 2.

Paso 2. Se busca el pivote usando

$$\frac{v_\alpha^{(0)}}{u_{\alpha\beta}^{(0)}} = \min_{v_i^{(0)}<0} \frac{v_i^{(0)}}{u_{i\beta}^{(0)}} = \min \left\{ \frac{-1}{1} \right\} = -1, \tag{6.78}$$

lo que implica $x_\alpha = x_5$.

Paso 3. Se realiza la pivotación para obtener $\mathbf{Z}^{(1)}$, que se muestra en la tabla 6.6. Nótese que tras la pivotación todos los coeficientes de $\mathbf{v}$, en la primera columna, son no negativos.

Paso 4. Como no se verifica la condición de solución, dado que no se cumple que $\mathbf{w} \geq \mathbf{0}$, se procede a encontrar la variable saliente usando

$$w_\beta^{(t)} = \min_{w_j^{(t)}<0} w_j^{(t)} = \min\{-3\} = -3 \tag{6.79}$$

lo que lleva a $x_\beta = x_1$.

Paso 5. Como todas las entradas de u en la columna asociada con x_1 son no negativas, el problema es no acotado. La dirección de descenso es

$$d = (0, 0, 5, 0, 1)^T$$

∎

Tabla 6.6: Tabla inicial con las matrices $u^{(0)}, \mathbf{w}^{(0)}, \mathbf{v}^{(0)}$, y $\mathbf{U}^{(0)}$ y tabla transformada después de la primera pivotación (Ejemplo 6.3)

$\mathbf{Z}^{(0)}$				$\mathbf{Z}^{(1)}$			
	1	x_1	x_2		1	x_5	x_2
Z	0	–3	–1	Z	–3	–3	2
x_3	2	3	–2	x_3	5	3	–5
x_4	0	0	1	x_4	0	0	1
x_5	–1	**1**	1	x_1	1	1	–1

Ejemplo 6.4 (ejemplo no factible). Considérese el siguiente PPL. Minimizar

$$Z = -3x_1 - x_2 \tag{6.80}$$

sujeto a

$$\begin{array}{rrcr} -3x_1 & +2x_2 & \leq & 2 \\ x_1 & +x_2 & \leq & 6 \\ x_1 & & \leq & 3 \\ 2x_1 & -x_2 & \leq & 4 \\ & -x_2 & \leq & 0 \\ -x_1 & -x_2 & \leq & -1 \\ x_1 & +x_2 & \leq & 0 \end{array} \tag{6.81}$$

con

$$x_1, x_2 \geq 0 \tag{6.82}$$

Mediante el algoritmo del MS se obtienen los siguientes resultados:

Paso 1. La tabla 6.7 muestra los valores iniciales de $\mathbf{Z}^{(0)}$ con las matrices $u^{(0)}, \mathbf{w}^{(0)}, \mathbf{v}^{(0)}$, y $\mathbf{U}^{(0)}$, donde se puede ver que algunos elementos en la primera columna son negativos. Por tanto, se necesita transformarlos en valores no negativos.

Como ninguna columna β verifica

$$\left\{ \begin{array}{lll} u_{i\beta}^{(0)} > 0 & \text{si} & v_i^{(0)} < 0 \\ u_{i\beta}^{(0)} \geq 0 & \text{si} & v_i^{(0)} \geq 0 \end{array} \right. \quad \forall i$$

se introduce la variable artificial $x_{10} > 0$, se modifica la función objetivo añadiendo el término $20x_{10}$, se suma el término x_{10} a todas las restricciones (variables básicas), y se hace $x_\beta = x_{10}$.

Paso 2. Se busca el pivote usando

$$\frac{v_\alpha^{(0)}}{u_{\alpha\beta}^{(0)}} = \min_{v_i^{(0)}<0} \frac{v_i^{(0)}}{u_{i\beta}^{(0)}} = \min \left\{ \frac{-1}{1} \right\} = -1 \tag{6.83}$$

lo que implica $x_\alpha = x_8$.

Tabla 6.7: Tabla inicial con las matrices $u^{(0)}, \mathbf{w}^{(0)}, \mathbf{v}^{(0)}$, y $\mathbf{U}^{(0)}$ y tabla transformada después de añadir la variable artificial x_{10} (Ejemplo 6.4)

$\mathbf{Z}^{(0)}$				$\mathbf{Z}^{(1)}$				
	1	x_1	x_2		1	x_1	x_2	x_{10}
Z	0	–3	–1	Z	0	–3	–1	20
x_3	2	3	–2	x_3	2	3	–2	1
x_4	6	–1	–1	x_4	6	–1	–1	1
x_5	3	–1	0	x_5	3	–1	0	1
x_6	4	–2	1	x_6	4	–2	1	1
x_7	0	0	1	x_7	0	0	1	1
x_8	–1	1	1	x_8	–1	1	1	**1**
x_9	0	–1	–1	x_9	0	–1	–1	1

Tabla 6.8: Tablas asociadas a las iteraciones 2 y 3 en que se muestran las matrices $u^{(0)}, \mathbf{w}^{(0)}, \mathbf{v}^{(0)}$, y $\mathbf{U}^{(0)}$ (Ejemplo 6.4)

$\mathbf{Z}^{(2)}$					$\mathbf{Z}^{(3)}$				
	1	x_1	x_2	x_8		1	x_9	x_2	x_8
Z	20	–23	–21	20	Z	17/2	23/2	2	17/2
x_3	3	2	–3	1	x_3	4	–1	–5	2
x_4	7	–2	–2	1	x_4	6	1	0	0
x_5	4	–2	–1	1	x_5	3	1	1	0
x_6	5	–3	0	1	x_6	$\frac{7}{2}$	$\frac{3}{2}$	3	$\frac{-1}{2}$
x_7	1	–1	0	1	x_7	$\frac{1}{2}$	$\frac{1}{2}$	1	$\frac{1}{2}$
x_{10}	1	–1	–1	1	x_{10}	$\frac{1}{2}$	$\frac{1}{2}$	0	$\frac{1}{2}$
x_9	1	**–2**	–2	1	x_1	$\frac{1}{2}$	$\frac{-1}{2}$	–1	$\frac{1}{2}$

Paso 3. Se realiza la pivotación para obtener $\mathbf{Z}^{(2)}$, que se muestra en la tabla 6.8 (donde el pivote aparece en negrita). Nótese que ahora todos los coeficientes de $\mathbf{v}$, en la primera columna, son no negativos.

Paso 4. Como no se verifica la condición de solución, puesto que no se cumple $\mathbf{w} \geq \mathbf{0}$, se busca la variable entrante usando

$$w_\beta^{(t)} = \min_{w_j^{(t)}<0} w_j^{(t)} = \min\{-23, -21\} = -23 \tag{6.84}$$

lo que lleva a $x_\beta = x_1$.

Paso 5. Se selecciona la variable saliente usando

$$\frac{v_\alpha^{(t)}}{u_{\alpha\beta}^{(t)}} = \max_{u_{i\beta}^{(t)}<0} \frac{v_i^{(t)}}{u_{i\beta}^{(t)}} = \max\left\{\frac{7}{-2}, \frac{4}{-2}, \frac{5}{-3}, \frac{1}{-1}, \frac{1}{-1}, \frac{1}{-2}\right\} = \frac{1}{-2} \tag{6.85}$$

lo que conduce a $x_\alpha = x_9$.

Paso 3. Se realiza la pivotación y se obtiene $\mathbf{Z}^{(3)}$, que se muestra en la tabla 6.8.

Paso 4. Como se satisface la condición de solución, pero $x_{10} = \frac{1}{2} > 0$, el problema no es factible. ■

6.3 El método del punto exterior

Debe tenerse en cuenta que el algoritmo del punto exterior es equivalente al algoritmo simplex dual.

Para comprender cómo funciona el método de punto exterior (MPE) y las diferencias esenciales con respecto al MS, descrito anteriormente, se empieza esta sección con un ejemplo sencillo. Considérese el siguiente PPL. Minimizar

$$Z = -3x_1 - x_2 \tag{6.86}$$

sujeto a

$$\begin{array}{rrcr} -3x_1 & +2x_2 & \le & 2 \\ x_1 & +x_2 & \le & 6 \\ x_1 & & \le & 3 \\ 2x_1 & -x_2 & \le & 4 \\ & -x_2 & \le & 0 \\ -x_1 & -x_2 & \le & -1 \end{array} \tag{6.87}$$

con

$$x_1, x_2 \ge 0 \tag{6.88}$$

El conjunto de puntos que satisfacen estas restricciones define la región factible; todos los demás puntos constituyen la región no factible (véase la figura 6.1).

En este caso, el MS empieza en el origen y se mueve, en las iteraciones sucesivas, a los vértices $(1,0)^T, (2,0)^T$, y $(3,2)^T$ antes de alcanzar el óptimo en el vértice $(3,3)^T$. Obsérvese que todos los puntos, excepto el inicial (el origen), pertenecen a la región factible del problema.

Por el contrario, el MPE trabaja fuera de la región factible. También empieza en el origen pero se mueve a los puntos $(6,0)^T$ y $(\frac{10}{3}, \frac{8}{3})^T$, antes de alcanzar el óptimo en el vértice $(3,3)^T$.

La interpretación geométrica de ambos métodos, dada en la figura 6.1, ayuda a la comprensión del MPE, cuyas bases teóricas se dan a continuación.

Como en el caso del MS, el MPE se aplica a un PPL con el siguiente formato. Minimizar

$$f(\mathbf{x}) = \mathbf{c}^T\mathbf{x} \tag{6.89}$$

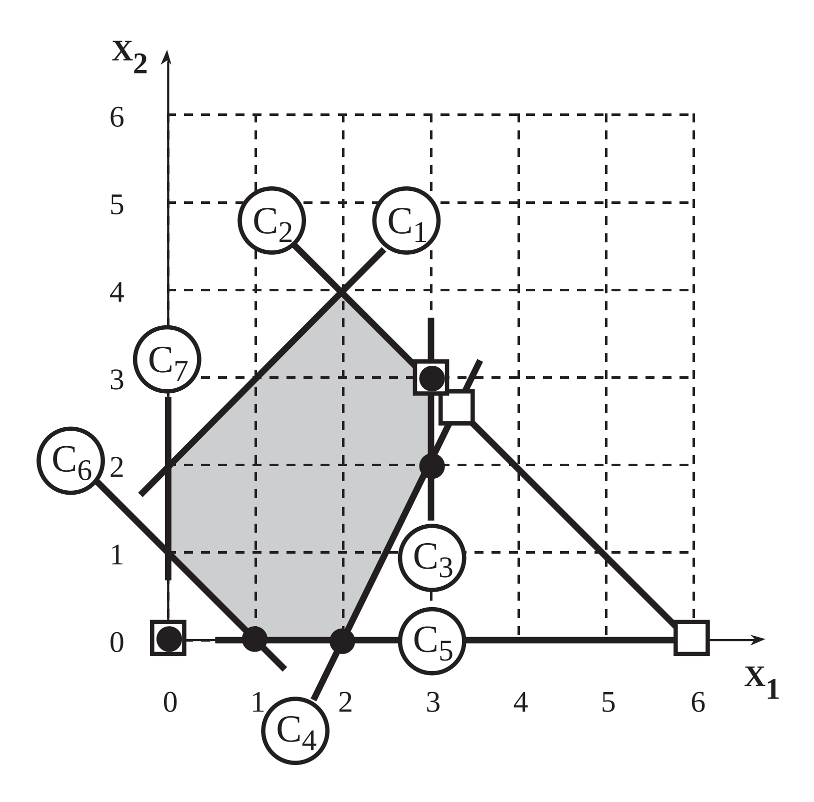


Figura 6.1: Gráfico que muestra las diferencias entre el MPE (cuadrados en blanco) y el MS (puntos negros).

sujeto a

$$\mathbf{A}\mathbf{x} = \mathbf{b} \tag{6.90}$$

y

$$\mathbf{x} \geq \mathbf{0} \tag{6.91}$$

donde $\mathbf{c} = (c_1, c_2, \ldots, c_n)^T$ es la matriz columna de los coeficientes de la función objetivo, $\mathbf{x} = (x_1, x_2, \ldots, x_n)^T$ es el vector columna de las variables iniciales, y $\mathbf{A}$ es una matriz $m \times n$ que contiene los coeficientes de las restricciones.

El MPE opera en tres etapas:

1. Una fase inicial, en que el conjunto original de restricciones de igualdad se transforma en otro equivalente de restricciones también de igualdad.

2. Una fase de regulación, en que los coeficientes de la función objetivo se convierten en valores no negativos

3. Una fase iterativa, en la que el valor de la función objetivo se mejora progresivamente hasta que se llega a la solución óptima, no se encuentran puntos factibles, o se establece que el problema es no acotado

6.3.1 Fase inicial

De forma análoga al MS, previamente descrito, el MPE incorpora una nueva variable Z, idéntica a la función objetivo del problema, y la restricción asociada

$$Z = 0x_0 + c_1x_1 + \cdots + c_nx_n \tag{6.92}$$

donde $x_0 = 1$.

El MPE empieza con el conjunto de restricciones (6.90) y (6.92) escritas en la forma

$$\begin{pmatrix} Z \\ -- \\ \mathbf{x}_B \end{pmatrix} = \begin{pmatrix} u_0^{(0)} & | & \mathbf{w}^{(0)T} \\ -- & + & -- \\ \mathbf{v}^{(0)} & | & \mathbf{U}^{(0)} \end{pmatrix} \begin{pmatrix} 1 \\ -- \\ \mathbf{x}_N \end{pmatrix} = \mathbf{Z}^{(0)} \begin{pmatrix} 1 \\ -- \\ \mathbf{x}_N \end{pmatrix} \tag{6.93}$$

donde $\mathbf{x}_B \cup \mathbf{x}_N$ es una partición del conjunto de variables $\{x_1, x_2, \ldots, x_n\}$, las matrices $\mathbf{v}^{(0)}$ y $\mathbf{U}^{(0)}$ se obtienen resolviendo (6.90) para $\mathbf{x}_B$ y empleando (6.12) y (6.13). Reemplazando $\mathbf{v}^{(0)}$ y $\mathbf{U}^{(0)}$ en (6.89) se obtiene:

$$u^{(0)} = \mathbf{c}_B^T\mathbf{v}^{(0)}, \quad \mathbf{w}^{(0)} = \mathbf{c}_B^T\mathbf{U}^{(0)} + \mathbf{c}_N^T \tag{6.94}$$

donde $\mathbf{c}_B^T$ y $\mathbf{c}_N^T$ son los coeficientes de la función objetivo asociados a $\mathbf{x}_B$ y $\mathbf{x}_N$, respectivamente.

El MPE convierte (6.93), en cada iteración, en un nuevo conjunto de restricciones con la misma estructura

$$\begin{pmatrix} Z^{(t)} \\ -- \\ \mathbf{x}_B^{(t)} \end{pmatrix} = \begin{pmatrix} u_0^{(t)} & | & \mathbf{w}^{(t)T} \\ -- & + & -- \\ \mathbf{v}^{(t)} & | & \mathbf{U}^{(t)} \end{pmatrix} \begin{pmatrix} 1 \\ -- \\ \mathbf{x}_N^{(t)} \end{pmatrix} = \mathbf{Z}^{(t)} \begin{pmatrix} 1 \\ -- \\ \mathbf{x}_N^{(t)} \end{pmatrix} \tag{6.95}$$

donde t indica el número de la iteración y $t = 0$ se usa para la iteración inicial.

La transformación más importante en el MPE es la de pivotación ya descrita anteriormente.

Las condiciones de solución óptima

$$\mathbf{w}^{(t)} \geq \mathbf{0}, \quad \mathbf{v}^{(t)} \geq \mathbf{0} \tag{6.96}$$

justifican el empleo de los dos pasos siguientes:

1. **Iteración reguladora**. Se usa una iteración reguladora para obtener $\mathbf{w}^{(t)} \geq \mathbf{0}$; esto es, para alcanzar una función objetivo con todos los coeficientes no negativos.

2. **Iteraciones propias**. Se lleva a cabo una transformación elemental con

   $$\mathbf{w}^{(t)} \geq \mathbf{0} \tag{6.97}$$

   para lograr

   $$\mathbf{v}^{(t)} \geq \mathbf{0} \tag{6.98}$$

   que conduce a la solución acotada; o se detecta la no acotación o la ausencia de solución.

6.3.2 Fase reguladora

En la iteración reguladora se transforma la función objetivo en una función con todos los coeficientes no negativos; esto es, se transforma $\mathbf{w}^{(0)}$ en $\mathbf{w}^{(1)} \geq \mathbf{0}$.
Se pueden tener una de estas dos situaciones:

1. Existe una restricción (fila), asociada a x_α, tal que

$$\left.\begin{array}{l} u_{\alpha j}^{(0)} < 0 \text{ si } w_j^{(0)} \geq 0 \\ u_{\alpha j}^{(0)} \leq 0 \text{ si } w_j^{(0)} < 0 \end{array}\right\} \forall j \neq 0 \tag{6.99}$$

2. No existe tal restricción. En este caso, se introduce la restricción adicional

$$x_1 + \cdots + x_n \leq M \Rightarrow x_{m+1} = M - x_1 - \cdots - x_n = \mathbf{u}_{m+1}^{(0)}\,\mathbf{x}_N \tag{6.100}$$

donde x_{m+1} es la nueva variable de holgura, M es un número mucho mayor que cualquier elemento de $\mathbf{U}^{(0)}$, que lógicamente verifica (6.99), y se toma $x_\alpha = x_{m+1}$.

En ambos casos se satisface (6.99) después de usar la transformación con el pivote $u_{\alpha\beta}^{(0)}$ dado por

$$\frac{w_\beta^{(0)}}{u_{\alpha\beta}^{(0)}} = \max_{w_j^{(0)}<0} \frac{w_j^{(0)}}{u_{\alpha j}^{(0)}} \tag{6.101}$$

Nota 6.3 *Con una selección apropiada de M, si el PPL tiene solución acotada, entonces $x_{m+1} > 0$.* ■

6.3.3 Detección de infactibilidad y de no acotación

En este apartado se explica cómo se puede detectar la no factibilidad o la no acotación.

1. **Situación de solución no acotada.** Esto sucede cuando se tiene $x_{m+1}^* = 0$ y $\mathbf{w}^{(t)} \geq \mathbf{0}$.

   Demostración. Es claro, pues $\mathbf{x}^* = \mathbf{v}^{(t)}$ (la restricción ficticia está activa) proporciona la solución del PPL y debido al comentario 6.3, el problema inicial (sin la restricción ficticia) es no acotado. ■

2. **Situación de solución no factible.** Esto ocurre cuando, para cada t, existe una fila i tal que

$$v_i^{(t)} < 0, \;\; u_{ij}^{(t)} \leq 0, \forall j \tag{6.102}$$

   Demostración. Como (i) $x_{B_i}^{(t)} = v_i^{(t)} + \sum_j u_{ij}^{(t)} x_{N\,j}^{(t)}$, (ii) $v_i^{(t)}$ y (iii) todos los elementos de $u_{ij}^{(t)}$ son no positivos, entonces, $x_{B_i}^{(t)}$ ha de ser negativo, para cualesquiera valores factibles de $\mathbf{x}_N^{(t)}$. Por tanto, no existen soluciones factibles. ■

6.3.4 Fase de iteraciones estándar

En esta etapa, se usa la transformación (6.24) para obtener $\mathbf{Z}^{(1)}, \ldots, \mathbf{Z}^{(t)}$, de forma que

1. Se consigue un sistema equivalente de restricciones:

$$\mathbf{x}_B^{(0)} = \mathbf{v}^{(0)} + \mathbf{U}^{(0)}\, \mathbf{x}_N^{(0)} \Leftrightarrow \mathbf{x}_B^{(t)} = \mathbf{v}^{(t)} + \mathbf{U}^{(t)}\, \mathbf{x}_N^{(t)}, \forall t \geq 1 \tag{6.103}$$

2. Se sigue cumpliendo la condición

$$\mathbf{w}^{(t)} \geq \mathbf{0} \tag{6.104}$$

3. El valor de la función objetivo no aumenta:

$$u_0^{(1)} \geq \cdots \geq u_0^{(t)} \tag{6.105}$$

4. Los pivotes se eligen para maximizar la disminución de $u_0^{(t)}$.

De esta manera, en cada iteración, la resolución del PPL es equivalente a minimizar Z sujeto a

$$\begin{array}{rcl} \mathbf{x}_B & = & \mathbf{v}^{(t)} + \mathbf{U}^{(t)}\mathbf{x}_N \\ \mathbf{x}_N & \geq & \mathbf{0} \\ \mathbf{x}_B & \geq & \mathbf{0} \end{array} \tag{6.106}$$

En la fase de iteraciones estándar, el MPE prosigue así:

1. Cada iteración tiene asociado un punto no factible que se obtiene dando el valor cero al conjunto de variables no básicas, $\mathbf{x}_N$. Por tanto, como $\mathbf{x}_N^{(t)} = \mathbf{0}$, $u_0^{(t)}$ siempre toma el valor de la función objetivo, y $\mathbf{x}_B$, el valor de las variables básicas.

2. El fundamento del MPE consiste en cambiar de modo adecuado el conjunto de variables básicas y no básicas en los pasos sucesivos del método. En cada iteración, una variable básica, x_α, y una no básica, x_β, intercambian sus papeles. Este cambio se lleva a cabo de modo que el nuevo conjunto de restricciones equivale al antiguo, en el sentido de que tienen las mismas soluciones. Se necesita por tanto

   (a) Establecer una regla para elegir la variable x_α que deja el conjunto $\mathbf{x}_B$ y entra en el conjunto $\mathbf{x}_N$

   (b) Establecer una regla para seleccionar la variable x_β que entra en el conjunto $\mathbf{x}_B$ y deja el conjunto $\mathbf{x}_N$

3. La selección de las variables entrante y saliente se hace disminuyendo el valor de la función objetivo y buscando obtener un punto factible.

Una vez que las variables saliente y entrante han sido seleccionadas, se debe transformar el sistema de restricciones en uno equivalente, usando la transformación de pivotación.

Selección de la variable saliente $\mathbf{x}_B$

De acuerdo con los resultados previos, si al alcanzar un estado t, la solución no se ha conseguido aún, y la infactibilidad o la no acotación no se han detectado, deben existir índices i y j tales que $v_i^{(t)} < 0$ y $u_{ij}^{(t)} > 0$.

La variable saliente x_α que deja $\mathbf{x}_B$ es la que satisface

$$v_\alpha^{(t)} = \min_{v_i^{(t)}<0} v_i^{(t)} \tag{6.107}$$

Nótese que, según la transformación de pivotación, el nuevo valor de $v_\alpha^{(t)}$ es

$$v_\alpha^{(t+1)} = -\frac{v_\alpha^{(t)}}{u_{\alpha\beta}} \geq 0 \tag{6.108}$$

De este modo, y según nuestras hipótesis previas, se obtiene

$$u_0^{(0)} \geq \cdots \geq u_0^{(t)} \Rightarrow Z^{(0)} \geq \cdots \geq Z^{(t)}$$

una relación que, interpretada en sentido geométrico, proporciona una sucesión de puntos $\mathbf{x}^{(0)}, \ldots, \mathbf{x}^{(t)}$, que se aproximan a los hiperplanos $f(\mathbf{x}) = f(\mathbf{x}^{(t)})$ en el punto de contacto $\mathbf{x}^* \in X$. Esto justifica el uso del término *punto exterior*, para identificar este método.

Selección de la variable entrante a $\mathbf{x}_B$

Dada la variable saliente x_α, el pivote $u_{\alpha\beta}^{(t)}$ de la transformación (6.24) se selecciona para que $\mathbf{w}^{(t)}$ satisfaga (6.104). Así, la columna pivote β y el valor del pivote $u_{\alpha\beta}^{(t)}$ se obtienen mediante

$$\Lambda_\beta^{(t)} = \frac{w_\beta^{(t)}}{u_{\alpha\beta}^{(t)}} = \min_{u_{\alpha j}^{(t)}>0} \frac{w_j^{(t)}}{u_{\alpha j}^{(t)}} \geq 0 \tag{6.109}$$

donde $\Lambda_\beta^{(t)}$ es la *razón función-pivote*. Lo anterior asegura que $\mathbf{w}^{(t+1)} \geq \mathbf{0}$. Por tanto, la variable asociada a la columna β abandona el conjunto $\mathbf{x}_N$.

Una justificación de (6.109) es la siguiente. De acuerdo a la transformación de pivotación (6.24), las nuevas variables de $w_j^{(t+1)}$ deben ser no negativas para todo j:

$$w_\beta^{(t+1)} = \frac{w_\beta^{(t)}}{u_{\alpha\beta}^{(t)}} \geq 0 \tag{6.110}$$

$$w_j^{(t+1)} = w_j^{(t)} - \Lambda_\beta^{(t)} u_{\alpha j}^{(t)} \geq 0; \ \ \forall j \neq \beta \tag{6.111}$$

La condición (6.110) se cumple por (6.109). Para comprobar que (6.111) también se cumple, se consideran dos posibilidades:

1. Si $u_{\alpha j}^{(t)} < 0$, entonces (6.111) es cierta, pues $w_j^{(t)}$ y $\Lambda_\beta^{(t)}$ son no negativos.

2. Si $u_{\alpha j}^{(t)} > 0$, entonces por la definición de mínimo, se tendrá

$$\Lambda_\beta^{(t)} \leq \frac{w_j^{(t)}}{u_{\alpha j}^{(t)}} \Leftrightarrow \Lambda_\beta^{(t)} u_{\alpha j}^{(t)} \leq w_j^{(t)} \tag{6.112}$$

lo que implica (6.111).

6.3.5 El algoritmo MPE

A continuación se desarrolla el algoritmo del MPE (método del punto exterior) que trabaja con la matriz $\mathbf{U}^{(t)}$.

Algoritmo 6.2 (algoritmo MPE)

- **Entrada.** los valores iniciales de, $\{\mathbf{U}^{(0)}, \mathbf{v}^{(0)}, \mathbf{w}^{(0)}, u^{(0)}\}$ del PPL, incluyendo la función objetivo, y las restricciones del problema.

- **Salida.** La solución del problema de minimización o un mensaje indicando la no factibilidad o la carencia de solución (solución no acotada).

Iteración reguladora

Paso 1 (iniciación). Si $\mathbf{w}^{(0)} \geq \mathbf{0}$, ir al paso 4. Si no, transformar la función objetivo de tal forma que sus coeficientes sean negativos. Con este fin, comprobar si existe alguna restricción α que verifique la condición

$$\left.\begin{array}{l} u_{\alpha j}^{(0)} < 0 \text{ si } w_j^{(0)} \geq 0 \\ u_{\alpha j}^{(0)} \leq 0 \text{ si } w_j^{(0)} < 0 \end{array}\right\} \forall j \neq 0 \tag{6.113}$$

Si esto es así, se continúa con el paso 2. Si no, se introduce la restricción artificial

$$x_{m+1} = M - x_1 - \cdots - x_n \tag{6.114}$$

y se elige $x_\alpha = x_{m+1}$.

Paso 2 (búsqueda del pivote). Encontrar la columna pivote β de la fila α usando

$$\frac{w_\beta^{(0)}}{u_{\alpha\beta}^{(0)}} = \max_{w_j^{(0)} < 0} \frac{w_j^{(0)}}{u_{\alpha j}^{(0)}} \tag{6.115}$$

e ir al paso 3.

Iteraciones estándar

Paso 3 (pivotación). Realizar la transformación de pivotación

$$z_{ij}^{(t+1)} = \left\{ \begin{array}{ll} \dfrac{z_{i\beta}^{(t)}}{z_{\alpha\beta}^{(t)}}, & \text{si} \quad i \neq \alpha, \quad j = \beta \\ z_{ij}^{(t)} - \dfrac{z_{\alpha j}^{(t)}}{z_{\alpha\beta}^{(t)}} z_{i\beta}^{(t)} & \text{si} \quad i \neq \alpha, \quad j \neq \beta \\ \dfrac{1}{z_{\alpha\beta}^{(t)}} & \text{si} \quad i = \alpha, \quad j = \beta \\ -\dfrac{z_{\alpha j}^{(t)}}{z_{\alpha\beta}^{(t)}} & \text{si} \quad i = \alpha, \quad j \neq \beta \end{array} \right\} \tag{6.116}$$

Paso 4 (selección de la variable saliente x_α). Si se cumplen la condición de solución

$$\mathbf{w}^{(t)} \geq \mathbf{0}, \quad \mathbf{v}^{(t)} \geq 0 \tag{6.117}$$

y la condición de no acotación $x^*_{m+1} = 0$, la solución del problema es no acotada. Si sólo se cumple la condición de solución, ésta es

$$\mathbf{x}^*_B = \mathbf{v}^{(t)}; \; \mathbf{x}^*_N = \mathbf{0}; \; Z = u_0^{(t)}, \tag{6.118}$$

y el algoritmo concluye. En otro caso, la variable x_α que deja $\mathbf{x}_B$ se selecciona usando

$$v_\alpha^{(t)} = \min_{v_i^{(t)} < 0} v_i^{(t)} \tag{6.119}$$

Si ninguna de estas condiciones se verifica, se continúa en el paso 5.

Paso 5 (selección de la variable entrante x_β). Se selecciona la variable x_β que abandona $\mathbf{x}_N$ usando

$$\frac{w_\beta^{(t)}}{u_{\alpha\beta}^{(t)}} = \min_{u_{\alpha j}^{(t)} > 0} \frac{w_j^{(t)}}{u_{\alpha j}^{(t)}} \tag{6.120}$$

Si se cumplen las condiciones de solución no factible

$$v_i^{(t)} < 0, u_{ij}^{(t)} \leq 0, \forall j \tag{6.121}$$

el algoritmo concluye dado que el problema es infactible. En otro caso se vuelve al paso 3. ■

6.3.6 Algunos ejemplos ilustrativos

En este sección se ilustra el MPE con varios ejemplos.

Tabla 6.9: Tabla inicial con las matrices $u^{(0)}, \mathbf{w}^{(0)}, \mathbf{v}^{(0)}$, y $\mathbf{U}^{(0)}$

$\mathbf{Z}^{(0)}$			
	x_0	x_1	x_2
Z	0	–3	–1
x_3	2	3	–2
x_4	6	**–1**	–1
x_5	3	–1	0
x_6	4	–2	1
x_7	0	0	1
x_8	–1	1	1

Ejemplo 6.5 (primer ejemplo). Examínese de nuevo el PPL siguiente. Minimizar

$$Z = -3x_1 - x_2 \tag{6.122}$$

sujeto a

$$\begin{array}{rrcr} -3x_1 & +2x_2 & \leq & 2 \\ x_1 & +x_2 & \leq & 6 \\ x_1 & & \leq & 3 \\ 2x_1 & -x_2 & \leq & 4 \\ & -x_2 & \leq & 0 \\ -x_1 & -x_2 & \leq & -1 \end{array} \tag{6.123}$$

con

$$x_1, x_2 \geq 0 \tag{6.124}$$

Mediante el MPE se obtienen los siguientes resultados:

Paso 1. La tabla 6.9 (donde los pivotes aparecen en negrita) muestra los valores iniciales con las matrices $u^{(0)}, \mathbf{w}^{(0)}, \mathbf{v}^{(0)}$, y $\mathbf{U}^{(0)}$, donde se puede ver que algunos de los elementos en la primera fila son negativos. En consecuencia, se necesita transformarlos a valores no negativos. Como la restricción asociada a la variable x_4 verifica la condición de regulación (6.113), se hace $x_\alpha = x_4$ y se continúa con el paso 2.

Paso 2. Como

$$\max\left(\frac{-3}{-1}, \frac{-1}{-1}\right) = 3$$

el pivote es -1, que corresponde a $x_\beta = x_1$.

Paso 3. Se realiza la pivotación (6.49) y se obtiene la tabla 6.10 donde se muestra $[\mathbf{Z}^{(1)}]$ (los pivotes aparecen en negrita).

Tabla 6.10: Tabla al final del paso 3 para las iteraciones 1, 2 y 3

$\mathbf{Z}^{(1)}$				$\mathbf{Z}^{(2)}$				$\mathbf{Z}^{(3)}$			
	x_0	x_4	x_2		x_0	x_4	x_6		x_0	x_4	x_5
Z	–18	3	2	Z	$\frac{-38}{3}$	$\frac{5}{3}$	$\frac{2}{3}$	Z	–12	1	2
x_3	20	–3	–5	x_3	$\frac{20}{3}$	$\frac{1}{3}$	$\frac{-5}{3}$	x_3	5	2	–5
x_1	6	–1	–1	x_1	$\frac{10}{3}$	$\frac{-1}{3}$	$\frac{-1}{3}$	x_1	3	0	–1
x_5	–3	1	1	x_5	$\frac{-1}{3}$	$\frac{1}{3}$	$\mathbf{\frac{1}{3}}$	x_6	1	–1	3
x_6	–8	2	**3**	x_2	$\frac{8}{3}$	$\frac{-2}{3}$	$\frac{1}{3}$	x_2	3	–1	1
x_7	0	0	1	x_7	$\frac{8}{3}$	$\frac{-2}{3}$	$\frac{1}{3}$	x_7	3	–1	1
x_8	5	–1	0	x_8	5	–1	0	x_8	5	–1	0

Paso 4. Como no se cumple la condición de solución y

$$\min_{v_i^{(t)}<0} v_i^{(t)} = \min(-3, -8) = -8$$

la variable entrante es $x_\alpha = x_6$.

Paso 5. Como

$$\min_{u_{6j}^{(t)}>0} \frac{w_j^{(t)}}{u_{6j}^{(t)}} = \min\left(\frac{3}{2}, \frac{2}{3}\right) = \frac{2}{3}$$

el pivote es 3 que corresponde a $x_\beta = x_2$.

Paso 3. Se realiza la transformación de pivotación para obtener $\mathbf{Z}^{(2)}$, que se muestra en la tabla 6.10.

Paso 4. Como no se cumple la condición de solución y

$$\min_{v_i^{(t)}<0} v_i^{(t)} = \min\left(\frac{-1}{3}\right) = -\frac{1}{3}$$

la variable entrante es $x_\alpha = x_5$.

Paso 5. Como

$$\min_{u_{5j}^{(t)}>0} \frac{w_j^{(t)}}{u_{5j}^{(t)}} = \min\left(\frac{5/3}{1/3}, \frac{2/3}{1/3}\right) = 2$$

el pivote es $\frac{1}{3}$ que corresponde a $x_\beta = x_6$.

Paso 3. Mediante la pivotación se llega a $\mathbf{Z}^{(3)}$, que se muestra en la tabla 6.10.

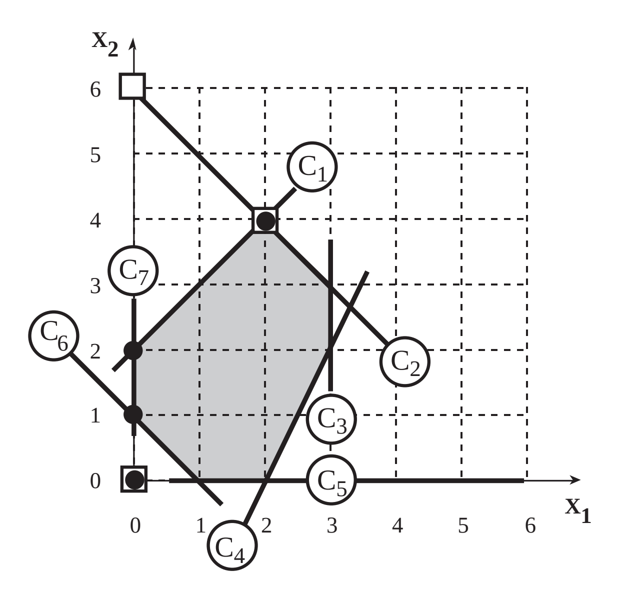


Figura 6.2: Gráfico con las diferencias entre el MPE (cuadrados blancos) y el MS (puntos negros).

Paso 4. Como todos los valores en la columna 1 son positivos, se ha encontrado la solución. El valor de la función objetivo es -12, que se alcanza en

$$x_1 = 3, x_2 = 3$$

La figura 6.2 muestra los vértices que ha "visitado" el MS (puntos negros) y el MPE (cuadrados blancos).

■

Ejemplo 6.6 (otro ejemplo). Considérese de nuevo el siguiente PPL. Minimizar

$$Z = -2x_1 - 4x_2 - 3x_3 + x_4 - 2x_5 - x_6 - 2x_7 - 2x_8 \tag{6.125}$$

sujeto a

$$\begin{array}{rrrrrrrrcr} 5x_1 & +4x_2 & +3x_3 & +2x_4 & +x_5 & +2x_6 & +3x_7 & +4x_8 & \leq & 17 \\ 2x_1 & -x_2 & +4x_3 & -2x_4 & +3x_5 & +3x_6 & +x_7 & -x_8 & \leq & 11 \\ 3x_1 & +2x_2 & +3x_3 & -2x_4 & +x_5 & -x_6 & +2x_7 & -x_8 & \leq & 8 \\ x_1 & -x_2 & +2x_3 & +2x_4 & -x_5 & & +x_7 & +x_8 & \leq & 4 \\ 4x_1 & +3x_2 & +x_3 & -x_4 & -2x_5 & +x_6 & +4x_7 & +3x_8 & \leq & 14 \end{array} \tag{6.126}$$

con

$$x_1, \ldots, x_8 \geq 0. \tag{6.127}$$

Tabla 6.11: Tabla inicial con las matrices $u^{(0)}, \mathbf{w}^{(0)}, \mathbf{v}^{(0)}$, y $\mathbf{U}^{(0)}$

	$\mathbf{Z}^{(0)}$								
	x_0	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8
Z	0	–2	–4	–3	1	–2	–1	–2	–2
x_9	17	–5	–4	–3	–2	**–1**	–2	–3	–4
x_{10}	11	–2	1	–4	2	–3	–3	–1	1
x_{11}	8	–3	–2	–3	2	–1	1	–2	1
x_{12}	4	–1	1	–2	–2	1	0	–1	–1
x_{13}	14	–4	–3	–1	1	2	–1	–4	–3

Mediante el MPE se llega a los siguientes resultados:

Paso 1. La tabla 6.11 (donde los pivotes aparecen en negrita) muestra los valores iniciales. Como algunos de los elementos en la primera fila son negativos, se necesita transformarlos en valores no negativos. Como la restricción asociada a la variable x_9 cumple la condición de regulación (6.113), se toma $x_\alpha = x_9$.

Paso 2. Como

$$\max\left(\frac{-2}{-5}, \frac{-4}{-4}, \frac{-3}{-3}, \frac{-2}{-1}, \frac{-1}{-2}, \frac{-2}{-3}, \frac{-2}{-4}\right) = 2$$

el pivote es -1, que corresponde a la columna de x_5 y a la fila de x_9.

Paso 3. Se realiza la transformación de pivotación y se obtiene la tabla 6.12 (donde los pivotes aparecen de nuevo en negrita).

Paso 4. Como no se cumple la condición de solución y

$$\min_{v_i^{(t)}<0} v_i^{(t)} = \min\left(-40, -9\right) = -40$$

la variable entrante es $x_\alpha = x_{10}$.

Paso 5. Como

$$\min_{u_{10j}^{(t)}>0} \frac{w_j^{(t)}}{u_{10j}^{(t)}} = \min\left(\frac{8}{13}, \frac{4}{13}, \frac{3}{5}, \frac{5}{8}, \frac{2}{3}, \frac{3}{3}, \frac{4}{8}, \frac{6}{13}\right) = \frac{4}{13}$$

el pivote es 13, que corresponde a la columna de x_2 y a la fila de x_{10}.

Paso 3. Se lleva a cabo la transformación de pivotación y se obtiene la tabla 6.13 (donde los pivotes están indicados en negrita).

Tabla 6.12: Tabla tras finalizar el paso 3 de la iteración 1

$\mathbf{Z}^{(1)}$									
	x_0	x_1	x_2	x_3	x_4	x_9	x_6	x_7	x_8
Z	–34	8	4	3	5	2	3	4	6
x_5	17	–5	–4	–3	–2	–1	–2	–3	–4
x_{10}	–40	13	**13**	5	8	3	3	8	13
x_{11}	–9	2	2	0	4	1	3	1	5
x_{12}	21	–6	–3	–5	–4	–1	–2	–4	–5
x_{13}	48	–14	–11	–7	–3	–2	–5	–10	–11

Tabla 6.13: Tabla tras finalizar el paso 3 en la iteración 2

$\mathbf{Z}^{(2)}$									
	x_0	x_1	x_{10}	x_3	x_4	x_9	x_6	x_7	x_8
Z	$\frac{-282}{13}$	4	$\frac{4}{13}$	$\frac{19}{13}$	$\frac{33}{13}$	$\frac{14}{13}$	$\frac{27}{13}$	$\frac{20}{13}$	2
x_5	$\frac{61}{13}$	–1	$\frac{-4}{13}$	$\frac{-19}{13}$	$\frac{6}{13}$	$\frac{-1}{13}$	$\frac{-14}{13}$	$\frac{-7}{13}$	0
x_2	$\frac{40}{13}$	–1	$\frac{1}{13}$	$\frac{-5}{13}$	$\frac{-8}{13}$	$\frac{-3}{13}$	$\frac{-3}{13}$	$\frac{-8}{13}$	–1
x_{11}	$\frac{-37}{13}$	0	$\frac{2}{13}$	$\frac{-10}{13}$	$\frac{36}{13}$	$\frac{7}{13}$	$\frac{33}{13}$	$\frac{-3}{13}$	**3**
x_{12}	$\frac{153}{13}$	–3	$\frac{-3}{13}$	$\frac{-50}{13}$	$\frac{-28}{13}$	$\frac{-4}{13}$	$\frac{-17}{13}$	$\frac{-28}{13}$	–2
x_{13}	$\frac{184}{13}$	–3	$\frac{-11}{13}$	$\frac{-36}{13}$	$\frac{49}{13}$	$\frac{7}{13}$	$\frac{-32}{13}$	$\frac{-42}{13}$	0

Paso 4. Como no se da la condición de solución y

$$\min_{v_i^{(t)}<0} v_i^{(t)} = \min\left(\frac{-37}{13}\right) = -\frac{37}{13}$$

la variable entrante es $x_\alpha = x_{11}$.

Paso 5. Como

$$\min_{u_{11j}^{(t)}>0} \frac{w_j^{(t)}}{u_{11j}^{(t)}} = \min\left(\frac{4/13}{2/13}, \frac{33/13}{36/13}, \frac{14/13}{7/13}, \frac{27/13}{33/13}, \frac{2}{3}\right) = \frac{2}{3}$$

el pivote es 3, que corresponde a la columna de x_8 y a la fila de x_{11}.

Paso 3. Se realiza la pivotación y se llega a la tabla 6.14.

Tabla 6.14: Tabla tras finalizar el paso 3 en la iteración 3

$\mathbf{Z}^{(3)}$									
	x_0	x_1	x_{10}	x_3	x_4	x_9	x_6	x_7	x_{11}
Z	$\frac{-772}{39}$	4	$\frac{8}{39}$	$\frac{77}{39}$	$\frac{9}{13}$	$\frac{28}{39}$	$\frac{5}{13}$	$\frac{22}{13}$	$\frac{2}{3}$
x_5	$\frac{61}{13}$	-1	$\frac{-4}{13}$	$\frac{-19}{13}$	$\frac{6}{13}$	$\frac{-1}{13}$	$\frac{-14}{13}$	$\frac{-7}{13}$	0
x_2	$\frac{83}{39}$	-1	$\frac{5}{39}$	$\frac{-25}{39}$	$\frac{4}{13}$	$\frac{-2}{39}$	$\frac{8}{13}$	$\frac{-9}{13}$	$\frac{-1}{3}$
x_8	$\frac{37}{39}$	0	$\frac{-2}{39}$	$\frac{10}{39}$	$\frac{-12}{13}$	$\frac{-7}{39}$	$\frac{-11}{13}$	$\frac{1}{13}$	$\frac{1}{3}$
x_{12}	$\frac{385}{39}$	-3	$\frac{-5}{39}$	$\frac{-170}{39}$	$\frac{-4}{13}$	$\frac{2}{39}$	$\frac{5}{13}$	$\frac{-30}{13}$	$\frac{-2}{3}$
x_{13}	$\frac{184}{13}$	-3	$\frac{-11}{13}$	$\frac{-36}{13}$	$\frac{49}{13}$	$\frac{7}{13}$	$\frac{-32}{13}$	$\frac{-42}{13}$	0

Paso 4. Como todos los valores en la columna 1 son positivos, se ha encontrado la solución. El valor de la función objetivo es $-772/39$, que se alcanza en

$$x_1 = 0, x_2 = \frac{83}{39}, x_3 = 0, x_4 = 0, x_5 = \frac{61}{13}, x_6 = 0, x_7 = 0, x_8 = \frac{37}{39}.$$

■

Ejemplo 6.7 (ejemplo no acotado). Considérese de nuevo el PPL. Minimizar

$$Z = -3x_1 - x_2 \tag{6.128}$$

sujeto a

$$\begin{array}{rrcr} -3x_1 & +2x_2 & \leq & 2 \\ & -x_2 & \leq & 0 \\ -x_1 & -x_2 & \leq & -1 \end{array} \tag{6.129}$$

con

$$x_1, x_2 \geq 0 \tag{6.130}$$

Mediante el MPE se obtienen los resultados siguientes:

Paso 1. La tabla 6.15 muestra los valores iniciales. Como algunos de los elementos en la primera fila son negativos, es preciso transformarlos en valores no negativos. Como ninguna restricción verifica la condición reguladora (6.113), se añade una restricción artificial, asociada a la variable básica $x_\alpha = x_6$ con $M = 20$.

Paso 2. Como

$$\max\left(\frac{-3}{-1}, \frac{-1}{-1}\right) = 3$$

Tabla 6.15: Tabla inicial con las matrices $u^{(0)}, \mathbf{w}^{(0)}, \mathbf{v}^{(0)}$, y $\mathbf{U}^{(0)}$

$\mathbf{Z}^{(0)}$				$\mathbf{Z}^{(0)}$			
	1	x_1	x_2		1	x_6	x_2
Z	0	–3	–1	Z	60	3	2
x_3	2	3	–2	x_3	62	–3	–5
x_4	0	0	1	x_4	0	0	1
x_5	–1	1	1	x_5	19	–1	0
x_6	20	**–1**	–1	x_1	20	–1	–1

el pivote es -1, que corresponde a la columna de x_1 y la fila de x_6.

Paso 3. Se realiza la transformación de pivotación y se obtiene $\mathbf{Z}^{(1)}$, que se muestra en la tabla 6.15.

Paso 4. Como se cumplen las condiciones de solución con $x_6 = 0$, el problema es no acotado.

■

Ejemplo 6.8 (PPL no factible revisado). Considérese de nuevo el PPL. Minimizar

$$Z = -3x_1 - x_2 \tag{6.131}$$

sujeto a

$$\begin{array}{rrcr} -3x_1 & +2x_2 & \leq & 2 \\ x_1 & +x_2 & \leq & 6 \\ x_1 & & \leq & 3 \\ 2x_1 & -x_2 & \leq & 4 \\ & -x_2 & \leq & 0 \\ -x_1 & -x_2 & \leq & -1 \\ x_1 & x_2 & \leq & 0 \end{array} \tag{6.132}$$

con

$$x_1 \geq 0, x_2 \geq 0 \tag{6.133}$$

El MPE nos conduce a los siguientes resultados:

Paso 1. La tabla 6.16 (donde los pivotes están en negrita) muestra los valores iniciales. Como algunos de los elementos en la primera fila son negativos, se necesita transformarlos en valores no negativos. Como la restricción asociada a x_4 verifica la condición reguladora (6.113), se hace $x_\alpha = x_4$.

Paso 2. Como

$$\max\left(\frac{-3}{-1}, \frac{-1}{-1}\right) = 1$$

Tabla 6.16: Tabla inicial con las matrices $u^{(0)}, \mathbf{w}^{(0)}, \mathbf{v}^{(0)}$, y $\mathbf{U}^{(0)}$

$\mathbf{Z}^{(0)}$				$\mathbf{Z}^{(1)}$			
	1	x_1	x_2		1	x_4	x_2
Z	0	– 3	–1	Z	–18	3	2
x_3	2	3	–2	x_3	20	–3	–5
x_4	6	**–1**	–1	x_1	6	–1	–1
x_5	3	–1	0	x_5	–3	1	1
x_6	4	–2	1	x_6	–8	2	**3**
x_7	0	0	1	x_7	0	0	1
x_8	–1	1	1	x_8	5	–1	0
x_9	0	–1	–1	x_9	–6	1	0

el pivote es -1 que corresponde a la columna de x_1 y a la fila de x_4.

Paso 3. Se realiza la pivotación y se encuentra $\mathbf{Z}^{(1)}$, que se muestra en la tabla 6.16.

Paso 4. Como no se cumple la condición de solución y

$$\min_{v_i^{(t)}<0} v_i^{(t)} = \min\left(-3, -8, -6\right) = -8$$

la variable saliente es $x_\alpha = x_6$.

Paso 5. Como

$$\min_{u_{6j}^{(t)}>0} \frac{w_j}{u_{6j}^{(t)}} = \min\left(\frac{3}{2}, \frac{2}{3}\right) = \frac{2}{3}$$

el pivote es 3, que corresponde a la columna de x_2 y a la fila de x_6.

Paso 3. Se lleva a cabo la transformación de pivotación y se obtiene $\mathbf{Z}^{(2)}$, que se muestra en la tabla 6.17.

Paso 4. Como no se cumple la condición de solución y

$$\min_{v_i^{(t)}<0} v_i^{(t)} = \min\left(\frac{-1}{3}, -6\right) = -6,$$

la variable saliente es $x_\alpha = x_9$.

Paso 5. Como

$$\min_{u_{9j}^{(t)}>0} \frac{w_j}{u_{9j}^{(t)}} = \min\left(\frac{5/3}{1}\right) = \frac{5}{3}$$

Tabla 6.17: Tabla de pivotaciones del ejemplo 6.8.

$\mathbf{Z}^{(2)}$				$\mathbf{Z}^{(3)}$				$\mathbf{Z}^{(4)}$			
	1	x_4	x_6		1	x_9	x_6		1	x_9	x_2
Z	$\frac{-38}{3}$	$\frac{5}{3}$	$\frac{2}{3}$	Z	$\frac{-8}{3}$	$\frac{5}{3}$	$\frac{2}{3}$	Z	0	3	2
x_3	$\frac{20}{3}$	$\frac{1}{3}$	$\frac{-5}{3}$	x_3	$\frac{26}{3}$	$\frac{1}{3}$	$\frac{-5}{3}$	x_3	2	–3	–5
x_1	$\frac{10}{3}$	$\frac{-1}{3}$	$\frac{-1}{3}$	x_1	$\frac{4}{3}$	$\frac{-1}{3}$	$\frac{-1}{3}$	x_1	0	–1	–1
x_5	$\frac{-1}{3}$	$\frac{1}{3}$	$\frac{1}{3}$	x_5	$\frac{5}{3}$	$\frac{1}{3}$	$\frac{1}{3}$	x_5	3	1	1
x_2	$\frac{8}{3}$	$\frac{-2}{3}$	$\frac{1}{3}$	x_2	$\frac{-4}{3}$	$\frac{-2}{3}$	$\mathbf{\frac{1}{3}}$	x_6	4	2	3
x_7	$\frac{8}{3}$	$\frac{-2}{3}$	$\frac{1}{3}$	x_7	$\frac{-4}{3}$	$\frac{-2}{3}$	$\frac{1}{3}$	x_7	0	0	1
x_8	5	–1	0	x_8	–1	–1	0	x_8	–1	–1	0
x_9	–6	**1**	0	x_4	6	1	0	x_4	6	1	0

el pivote es 1, que corresponde a la columna de x_4 y la fila de x_9.

Paso 3. Se realiza la transformación de pivotación para obtener $\mathbf{Z}^{(3)}$, que se muestra en la tabla 6.17.

Paso 4. Como no se cumple la condición de solución y

$$\min_{v_i^{(t)}<0} v_i^{(t)} = \min\left(\frac{-4}{3}, \frac{-4}{3}\right) = -\frac{4}{3}$$

la variable saliente es $x_\alpha = x_2$.

Paso 5. Como

$$\min_{u_{2j}^{(t)}>0} \frac{w_j}{u_{2j}^{(t)}} = \min\left(\frac{2/3}{1/3}\right) = \frac{2/3}{1/3}$$

el pivote es $\frac{1}{3}$, que corresponde a la columna de x_6 y la fila de x_2.

Paso 3. Se realiza la transformación de pivotación y se obtiene $\mathbf{Z}^{(4)}$, que se muestra en la tabla 6.17.

Paso 4. Como no se cumple la condición de solución y

$$\min_{v_i^{(t)}<0} v_i^{(t)} = \min(-1) = -1$$

la variable saliente es $x_\alpha = x_8$.

Paso 5. Como se cumple la condición de infactibilidad, pues todos los valores u_{8j} son no positivos, el problema no tiene solución factible.

■

Ejercicios

6.1 Determínese la región factible y sus vértices en el problema siguiente. Minimizar

$$Z = -4x_1 + 7x_2$$

sujeto a

$$\begin{array}{rrcl} x_1 & +x_2 & \geq & 3 \\ -x_1 & +x_2 & \leq & 3 \\ 2x_1 & +x_2 & \leq & 8 \\ & x_1,\, x_2 & \geq & 0 \end{array}$$

Obténgase la solución óptima mediante el MS y el MPE.

6.2 Pruébese que el problema siguiente no tiene un óptimo finito; empléese el MS y el MPE. Minimizar

$$Z = -6x_1 + 2x_2$$

sujeto a

$$\begin{array}{rrcr} -3x_1 & +x_2 & \leq & 6 \\ 3x_1 & +5x_2 & \geq & 15 \\ & x_1,\, x_2 & \geq & 0 \end{array}$$

6.3 Muéstrese, mediante el MS y el MPE, que el siguiente problema de maximización de

$$Z = -6x_1 + 2x_2$$

sujeto a

$$\begin{array}{rrcr} -3x_1 & +x_2 & \leq & 6 \\ 3x_1 & +5x_2 & \geq & 15 \\ & x_1,\, x_2 & \geq & 0 \end{array}$$

tiene un máximo infinito.

6.4 Resuélvase el dual del problema. Minimizar

$$Z = 5x_1 - 6x_2 + 7x_3 + x_4$$

sujeto a

$$\begin{array}{rrrrcr} x_1 & +2x_2 & -x_3 & -x_4 & = & -7 \\ 6x_1 & -3x_2 & +x_3 & +7x_4 & \geq & 14 \\ -2x_1 & -7x_2 & +4x_3 & +2x_4 & \leq & -3 \\ & & & x_1,\, x_2 & \geq & 0 \end{array}$$

6.5 Resuélvase el siguiente problema: Maximizar

$$Z = 240x_1 + 104x_2 + 60x_3 + 19x_4$$

sujeto a

$$\begin{array}{rcl} 20x_1 + 9x_2 + 6x_3 + x_4 & \leq & 20 \\ 10x_1 + 4x_2 + 2x_3 + x_4 & \leq & 10 \\ x_i & \geq & 0, \quad i = 1, 2, 3, 4 \end{array}$$

analizando y resolviendo su dual.

6.6 Determínese, usando el MS y el MPE, el valor máximo de la función objetivo

$$Z = 18x_1 + 4x_2 + 6x_3$$

sujeto a las restricciones

$$\begin{array}{rcl} 3x_1 + x_2 & \leq & -3 \\ 2x_1 + x_3 & \leq & -5 \\ x_i & \leq & 0, \text{ for } i = 1, 2, 3 \end{array}$$

6.7 Un productor de electricidad debe planificar su producción horaria de energía para maximizar sus beneficios por venta de la misma en un horizonte de 2 horas. Formúlese y resuélvase el PPL que consiste en maximizar los beneficios del productor si

(a) Se producen 5 unidades de energía antes del periodo de planificación.

(b) Los precios horarios de la energía son 6 y 2 unidades monetarias.

(c) La energía mínima que se puede producir en cada hora es 0 y la máxima 10 unidades.

(d) Las producciones de energía en dos horas consecutivas no pueden diferir más de 4 unidades.

(e) El coste de producción es 3 unidades monetarias por unidad de energía.

6.8 Resuélvase por el MS, el ejercicio 1.7, donde $c = 1$, y los cuatro puntos fijos son

$$(1, 0), (0, 1), (-1, 0), (0, -1)$$

Capítulo 7

Programación lineal entera-mixta

7.1 Introducción

Un problema de programación lineal entera-mixta (PPLEM) es un problema de programación lineal (PPL) en el que algunas de las variables son enteras. Si todas las variables enteras son binarias (0/1), el problema se denomina problema de programación lineal entera-mixta 0/1 (PPLEM 0/1). Si, por otra parte, todas las variables son enteras, el problema se denomina problema de programación lineal entera estricta (PPLEE). En ingeniería los problemas más frecuentes son los problemas de programación lineal entera-mixta. Estos problemas proporcionan un marco de modelado flexible y eficiente para formular y resolver muchos problemas de ingeniería. Este capítulo proporciona una introducción a la programación lineal entera-mixta y a las técnicas de solución asociadas a la misma. Un ejemplo del conjunto de soluciones factibles de un problema de programación lineal entera estricta se muestra en la figura 7.1, en la que los puntos factibles se indican mediante círculos.

Un PPLEM general se formula en forma estándar minimizando

$$Z = \sum_{j=1}^{n} c_j x_j$$

sujeto a

$$\begin{array}{rcll} \sum_{j=1}^{n} a_{ij} x_j & = & b_i; & i = 1, 2, \ldots, m \\ x_j & \geq & 0; & j = 1, 2, \ldots, n \\ x_j & \in & \mathbb{N}; & \text{para todos o algunos } j = 1, 2, \ldots, n \end{array}$$

donde $\mathbb{N}$ se emplea en este capítulo para referirse al conjunto $\{0, 1, 2, \ldots\}$.

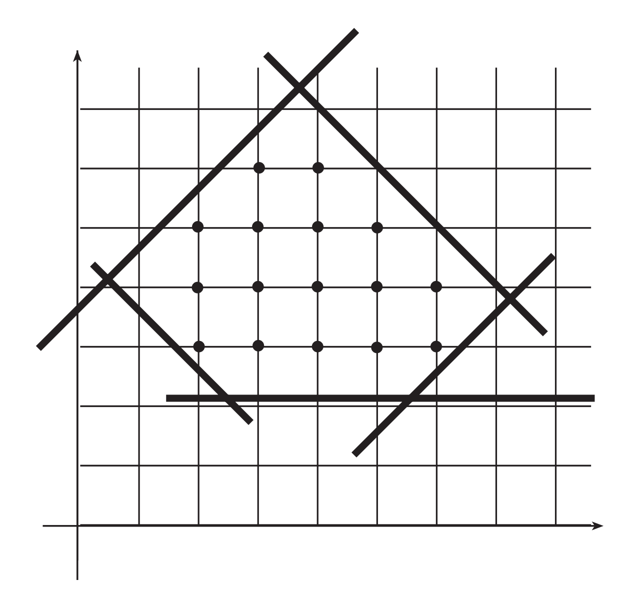


Figura 7.1: Conjunto de soluciones factibles de un PPLEE (puntos asociados a círculos).

Referencias relevantes para el estudio de PPLEMs son el manual clásico de Garfinkel y Nemhauser [47], la excelente monografía de Nemhauser y Wolsey [79], el manual con orientación práctica de Bradley et al. [15], y el manual introductorio de Rao [92].

En este capítulo se explican dos procedimientos de resolución de PPLEMs: La técnica de ramificación y acotación (RA) y la técnica de los cortes de Gomory (CG). La técnica RA es la utilizada más habitualmente por sus buenas propiedades computacionales. Sin embargo, recientemente se han desarrollado técnicas híbridas, denominadas de ramificación y cortes, que son más eficaces.

Las variables binarias son un mecanismo eficaz de modelado de no linealidades que aparecen con frecuencia en ingeniería. En el capítulo 13 se mostrarán algunos ejemplos, como:

1. Conjuntos alternativos de restricciones

2. Restricciones condicionales

3. Funciones discontinuas

4. Funciones lineales a tramos y no convexas

El modelado de no linealidades mediante el empleo de variables binarias se explica en detalle, mediante ejemplos, en el capítulo 13, Sección 13.2.9.

7.2 El método de ramificación y acotación

7.2.1 Introducción

El método RA resuelve PPLEMs resolviendo una secuencia ordenada de PPL, que se obtienen relajando las restricciones de integralidad y añadiendo restricciones adicionales. El número de restricciones adicionales crece a medida que el procedimiento de RA progresa. Estas restricciones permiten separar la región factible en subregiones complementarias.

El procedimiento RA establece inicialmente cotas inferior y superior del valor óptimo de la función objetivo. El mecanismo de ramificación aumenta progresivamente el valor de la cota superior y disminuye también progresivamente el valor de la cota superior. La diferencia entre estas cotas es una medida de la proximidad de la solución actual a la óptima, si ésta existe.

Al minimizar, se obtiene una cota inferior de la solución óptima relajando las restricciones de integralidad del PPLEM inicial y resolviendo el PPL resultante. De manera análoga, el valor de la función objetivo para cualquier solución del PPLEM original es una cota superior de la solución óptima.

7.2.2 El algoritmo de RA para un PPLEM

Entrada. Un PPLEM que ha de resolverse.

Salida. Su solución o un mensaje indicando que es infactible o que no está acotado.

Paso 1. (iniciación). Se establece una cota superior (∞) y una cota inferior ($-\infty$) de la solución óptima. Se resuelve el PPLEM inicial relajando las restricciones de integralidad. Si el problema relajado es infactible, el original también lo es y no hay solución. Si la solución obtenida satisface las condiciones de integralidad, es óptima. En cualquier otro caso, se actualiza el valor de la cota inferior con el valor de la función objetivo del problema relajado.

Paso 2. (ramificación). Empleando la variable x_k que ha de ser entera y no lo es, se generan mediante ramificación dos problemas. Si el valor de la variable que ha de ser entera x_k es $a.b$, donde a y b son sus partes entera y fraccional respectivamente, los problemas fruto de la ramificación son los siguientes. El primer problema es el PPLEM inicial relajado al que se la añade la restricción $x_k \leq a$; análogamente, el segundo es el PPLEM inicial relajado al que se le añade la restricción $x_k \geq a + 1$. Estos problemas se colocan ordenadamente en una lista de problemas a procesar que son resueltos secuencialmente o en paralelo. Obsérvese que la técnica de ramificación propuesta cubre completamente el espacio de soluciones.

Paso 3. (solución). Se resuelve el problema siguiente en la lista de problemas a procesar.

Paso 4. (actualización de cotas). Si la solución del problema actual satisface las condiciones de integralidad y el valor óptimo de su función objetivo es menor que la cota superior actual, la cota superior se actualiza al valor óptimo de la función objetivo del problema resuelto, y el minimizador actual se almacena como el mejor candidato a minimizador del problema original. Si, por el contrario, la solución obtenida no satisface las restricciones de integralidad y el valor de la correspondiente función objetivo está entre las cotas inferior y superior, se actualiza el valor de la cota inferior al valor de la función objetivo del problema resuelto y se procede a ramificar. Los problemas generados en el proceso de ramificación se añaden a la lista de problemas que han de resolverse.

Paso 5. (poda). Si la solución del problema actual cumple las restricciones de integralidad, no ha lugar a ramificaciones adicionales relacionadas con esa solución. Se dice que la rama se poda por razones de integralidad. Si, por otra parte, la solución no satisface las condiciones de integralidad y además el valor de la función objetivo del problema resuelto es mayor que la cota superior, no es posible obtener soluciones mediante ramificaciones adicionales de esa rama. Se dice que la rama se poda por cotas. Si, finalmente, el problema es infactible, no ha lugar a ramificaciones adicionales empleando esa rama. Se dice que la rama se poda por infactibilidad.

Paso 6. (optimalidad). Si la lista de problemas a procesar no está vacía, se continúa con el paso 3. Si, por el contrario, está vacía, el procedimiento concluye. Si en este caso, existe un candidato a minimizador, este candidato es el minimizador; si no existe, el problema es infactible.

El algoritmo de ramificación y acotación devuelve la solución óptima o notifica la infactibilidad bien en el paso 1 ó en el paso 6.

El proceso de ramificación concluye por una de las siguientes tres razones:

1. El problema considerado es infactible
2. La solución obtenida satisface las condiciones de integralidad
3. La cota inferior obtenida es superior a la cota superior disponible

Por tanto, la rama correspondiente se poda por infactibilidad, por integralidad o por cotas.

7.2.3 Estrategias de ramificación y procesamiento

Cualquier variable que deba ser entera pero que no lo sea en la solución actual, es una variable candidata para ramificación. Cuál escoger es una cuestión no trivial cuya respuesta ha de basarse en la estructura del problema.

Los problemas almacenados para ser procesados pueden tratarse mediante estrategias *en anchura*, *en profundidad* o *mixtas*. La figura 7.2 ilustra las

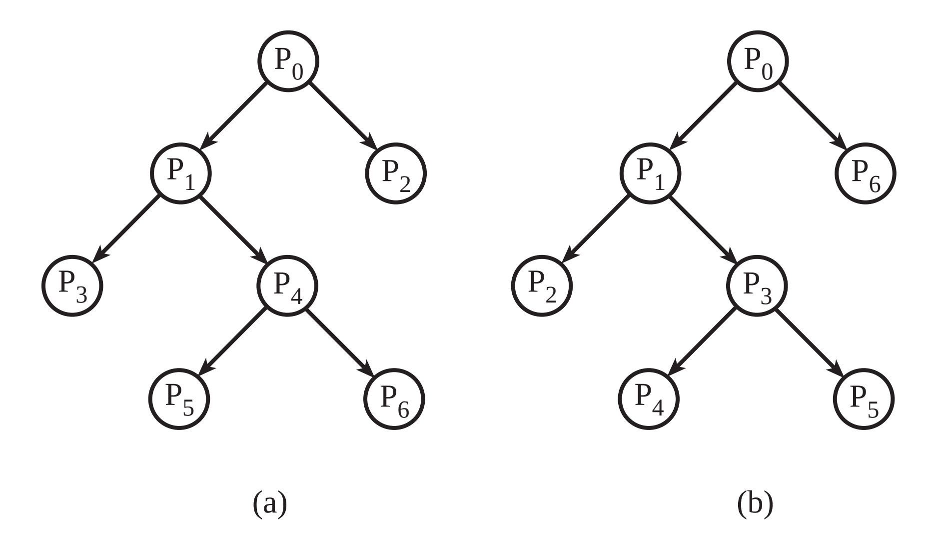


Figura 7.2: Ilustración de estrategias de (a) 'búsqueda en anchura', y (b) 'búsqueda en profundidad'.

dos primeras alternativas. Normalmente el conocimiento técnico del problema permite establecer una estrategia, la otra o una estrategia mixta.

Una estrategia de procesado en profundidad origina rápidamente problemas fuertemente restringidos que producen buenas cotas superiores e inferiores. Da lugar asimismo a problemas infactibles y por tanto a una deseable eliminación de ramas. Por el contrario, una estrategia en anchura permite tratar problemas muy similares, de lo que pueden desprenderse ventajas computacionales. Las técnicas de procesado paralelo son aplicables a cualquiera de las dos estrategias.

Ejemplo 7.1 (método de ramificación y acotación: problema entero estricto). Considérese el siguiente PPLE. Minimizar

$$Z = -x_1 - x_2$$

sujeto a

$$\begin{array}{rrcl} -x_1 & & \leq & 0 \\ 2x_1 & -2x_2 & \leq & 1 \\ & 2x_2 & \leq & 9 \\ & x_1, x_2 & \in & \mathbb{N} \end{array}$$

que es la región factible representada mediante círculos en la figura 7.3.

Paso 1. (iniciación). La cota superior inicial es $+\infty$ y la inferior $-\infty$. El problema relajado, denominado P_0, es el siguiente. Minimizar

$$Z = -x_1 - x_2$$

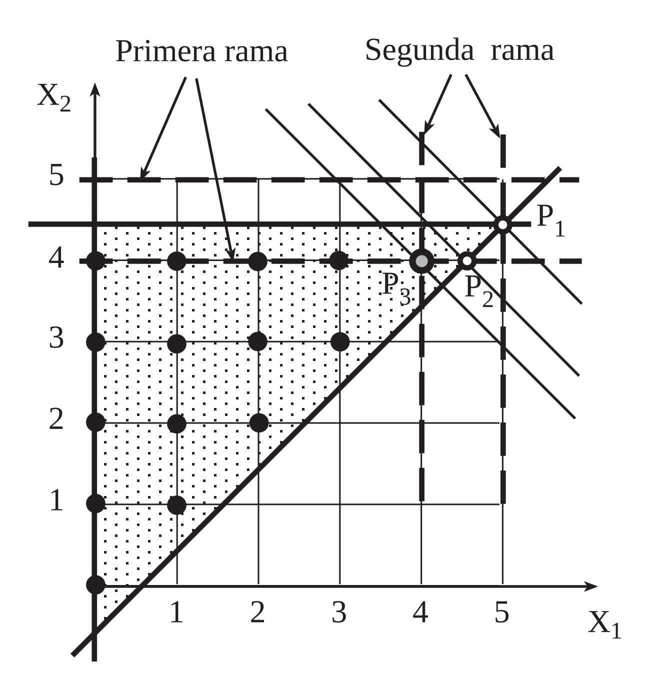


Figura 7.3: Ilustración gráfica del método de RA en el Ejemplo 7.1.

sujeto a

$$\begin{array}{rrcl} -x_1 & & \leq & 0 \\ 2x_1 & -2x_2 & \leq & 1 \\ & 2x_2 & \leq & 9 \end{array}$$

Su solución es

$$x_1 = 5, \quad x_2 = 4.5; \quad Z = -9.5$$

que es el punto P_1 en la figura 7.3. Esta solución no satisface las condiciones de integralidad ($x_2 \notin \mathbb{N}$). El valor de la función objetivo se emplea para actualizar la cota inferior de $-\infty$ a -9.5.

Paso 2. (ramificación). La variable que ha de ser entera x_2, mediante ramificación da lugar a los dos problemas siguientes. Las restricciones adicionales son $x_2 \leq 4$ y $x_2 \geq 5$, que se muestran en la figura 7.3 como líneas horizontales discontinuas.

Problema P_1. Minimizar

$$Z = -x_1 - x_2$$

sujeto a

$$\begin{array}{rrcl} -x_1 & & \leq & 0 \\ 2x_1 & -2x_2 & \leq & 1 \\ & 2x_2 & \leq & 9 \\ & x_2 & \leq & 4 \end{array}$$

Problema P_2. Minimizar

$$Z = -x_1 - x_2$$

sujeto a

$$\begin{array}{rrcl} -x_1 & & \leq & 0 \\ 2x_1 & -2x_2 & \leq & 1 \\ & 2x_2 & \leq & 9 \\ & x_2 & \geq & 5 \end{array}$$

Paso 3. (solución). La solución del problema P_1 es

$$x_1 = 4.5; \;\; x_2 = 4; \;\; Z = -8.5$$

que es el punto P_2 en la figura 7.3.

Paso 4. (actualización de cotas). Puesto que la solución obtenida no satisface las condiciones de integralidad ($x_1 \notin \mathbb{N}$), y que el valor de la función objetivo, -8.5, está entre los valores de las cotas inferior y superior, el valor actual de la cota inferior se actualiza de -9.5 a -8.5 (la solución óptima está por tanto entre -8.5 y ∞). El problema se vuelve a ramificar. La variable que deber ser entera pero no lo es, x_1, mediante ramificación, da lugar a los dos problemas siguientes. Las restricciones adicionales son $x_1 \leq 4$ y $x_1 \geq 5$, que se muestran en la figura 7.3 como líneas verticales a trazos.

Problema P_3. Minimizar

$$Z = -x_1 - x_2$$

sujeto a

$$\begin{array}{rrcl} -x_1 & & \leq & 0 \\ 2x_1 & -2x_2 & \leq & 1 \\ & 2x_2 & \leq & 9 \\ & x_2 & \leq & 4 \\ x_1 & & \leq & 4 \end{array}$$

Problema P_4. Minimizar

$$Z = -x_1 - x_2$$

sujeto a

$$\begin{array}{rrcl} -x_1 & & \leq & 0 \\ 2x_1 & -2x_2 & \leq & 1 \\ & 2x_2 & \leq & 9 \\ & x_2 & \leq & 4 \\ x_1 & & \geq & 5 \end{array}$$

Paso 5. (poda). No ocurre nada en este paso.

Paso 6. (control de optimalidad). Puesto que la lista de problemas a procesar no está vacía, se continúa con el problema P_2 en el paso 3.

Paso 3. (solución). El problema P_2 es infactible; por tanto, nada tiene lugar en este paso.

Paso 4. (actualización de cotas). Nada tiene lugar en este paso.

Paso 5. (poda). Puesto que se trata de un problema infactible, la rama se poda.

Paso 6. (control de optimalidad). Puesto que la lista de problemas no está vacía, se continúa con el problema P_3 en el paso 3.

Paso 3. (solución). La solución del problema P_3 es

$$x_1 = 4; \quad x_2 = 4; \quad z = -8$$

que es el punto P_3 en la figura 7.3.

Paso 4. (actualización de cotas). Puesto que la solución obtenida cumple las condiciones de integralidad ($x_1, x_2 \in \mathbb{N}$), y el valor de la función objetivo, -8, está entre las cotas inferior y superior, el valor de la cota superior se actualiza de $+\infty$ a -8, y el minimizador obtenido se almacena como mejor candidato.

Paso 5. (poda). Puesto que la solución actual satisface las condiciones de integralidad, no procede ramificar adicionalmente, y la rama se poda.

Paso 6. (control de optimalidad). Puesto que la lista de problemas a procesar no está vacía, se continúa en el paso 3.

Paso 3. (solución). El problema P_4 es infactible, por tanto, no tiene lugar nada en este paso.

Paso 4. (actualización de cotas). Nada tiene lugar en este paso.

Paso 5. (poda). Puesto que el problema es infactible, la rama correspondiente se poda.

Paso 6. (control de optimalidad). Puesto que la lista de problemas a procesar está vacía, y hay un candidato a minimizador, este candidato es la solución del problema original:

$$x_1^* = 4; \;\; x_2^* = 4; \;\; Z^* = -8$$

■

Ejemplo 7.2 (método de ramificación y acotación: problema entero-mixto). Considérese el siguiente PPLEM. Minimizar

$$Z = 3x_1 + 2x_2$$

sujeto a

$$\begin{array}{rrrrcl} x_1 & -2x_2 & +x_3 & & = & \dfrac{5}{2} \\ 2x_1 & +x_2 & & +x_4 & = & \dfrac{3}{2} \\ & & & x_1, x_2, x_3, x_4 & \geq & 0 \\ & & & x_2, x_3 & \in & \mathbb{N} \end{array}$$

Paso 1. (iniciación). La cota superior inicial es $+\infty$ y la inferior $-\infty$. El problema relajado, denominado P_0, es: Minimizar

$$Z = 3x_1 + 2x_2$$

sujeto a

$$\begin{array}{rrrrcl} x_1 & -2x_2 & +x_3 & & = & \dfrac{5}{2} \\ 2x_1 & +x_2 & & +x_4 & = & \dfrac{3}{2} \\ & & & x_1, x_2, x_3, x_4 & \geq & 0 \end{array}$$

La solución es

$$x_1 = x_2 = 0; \;\; x_3 = 2.5; \;\; x_4 = 1.5; \;\; Z = 0$$

Esta solución no satisface las condiciones de integralidad ($x_3 \notin \mathbb{N}$). El valor de la función objetivo se emplea para actualizar la cota inferior de $-\infty$ a 0.

Paso 2. (ramificación). La variable que ha de ser entera, x_3, mediante ramificación da lugar a los dos problemas siguientes. Las restricciones adicionales son $x_3 \leq 2$ y $x_3 \geq 3$.

Problema P_1. Minimizar

$$Z = 3x_1 + 2x_2$$

sujeto a

$$\begin{array}{rrrrcl} x_1 & -2x_2 & +x_3 & & = & \frac{5}{2} \\ 2x_1 & +x_2 & & +x_4 & = & \frac{3}{2} \\ & & x_3 & & \leq & 2 \\ & & & x_1, x_2, x_3, x_4 & \geq & 0 \end{array}$$

Problema P_2. Minimizar

$$Z = 3x_1 + 2x_2$$

sujeto a

$$\begin{array}{rrrrcl} x_1 & -2x_2 & +x_3 & & = & \frac{5}{2} \\ 2x_1 & +x_2 & & +x_4 & = & \frac{3}{2} \\ & & x_3 & & \geq & 3 \\ & & & x_1, x_2, x_3, x_4 & \geq & 0 \end{array}$$

Paso 3. (solución). La solución del problema P_1 es

$$x_1 = 0.5; \;\; x_2 = 0; \;\; x_3 = 2; \;\; x_4 = 0.5; \;\; Z = 1.5$$

Paso 4. (actualización de cotas). Puesto que la solución obtenida satisface las condiciones de integralidad ($x_2, x_3 \in \mathbb{N}$), y el valor de la función objetivo, 1.5, es menor que el valor actual de la cota superior, éste se actualiza de $+\infty$ a 1.5 (la solución óptima está por tanto entre 0 y 1.5), y el minimizador encontrado se almacena como mejor candidato.

Paso 5. (poda). Puesto que la solución obtenida satisface la condiciones de integralidad (para x_2 y x_3), la rama se poda. Se continúa con el paso 3.

Paso 3. (solución). La solución del problema P_2 es

$$x_1 = 0; \;\; x_2 = 0.25; \;\; x_3 = 3; \;\; x_4 = 1.25; \;\; Z = 0.5$$

Paso 4. (actualización de cotas). Puesto que esta solución no satisface las condiciones de integralidad ($x_2 \notin \mathbb{N}$) y el valor correspondiente de la función objetivo, 0.5, está entre las cotas interior y superior, la cota inferior se actualiza

de 0 a 0.5. Por tanto la solución óptima está en el intervalo $(0.5, 1.5)$. A continuación se procede a ramificar sobre la variable x_2, lo que da lugar a los dos problemas siguientes:

Problema P_3. Minimizar

$$Z = 3x_1 + 2x_2$$

sujeto a

$$\begin{array}{rrrrcl} x_1 & -2x_2 & +x_3 & & = & \frac{5}{2} \\ 2x_1 & +x_2 & & +x_4 & = & \frac{3}{2} \\ & & x_3 & & \geq & 3 \\ & x_2 & & & \leq & 0 \\ & & & x_1, x_2, x_3, x_4 & \geq & 0 \end{array}$$

Problema P_4. Minimizar

$$Z = 3x_1 + 2x_2$$

sujeto a

$$\begin{array}{rrrrcl} x_1 & -2x_2 & +x_3 & & = & \frac{5}{2} \\ 2x_1 & +x_2 & & +x_4 & = & \frac{3}{2} \\ & & x_3 & & \geq & 3 \\ & x_2 & & & \geq & 1 \\ & & & x_1, x_2, x_3, x_4 & \geq & 0 \end{array}$$

que se añaden a la lista de problemas a procesar. Obsérvese que las restricciones adicionales son respectivamente $x_2 \leq 0$ y $x_2 \geq 1$.

Paso 5. (poda). Nada tiene lugar en este paso, se continúa con el paso 6.

Paso 6. (control de optimalidad). Puesto que la lista de problemas a procesar no está vacía, se continúa con el paso 3.

Paso 3. (solución). El problema P_3 es infactible.

Paso 4. (actualización de cotas). Nada tiene lugar en este paso.

Paso 5. (poda). Puesto que el problema es infactible, se poda la rama correspondiente, y se continúa con el paso 6.

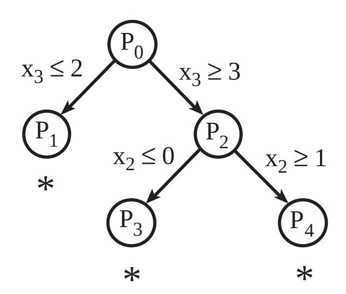


Figura 7.4: Ilustración del procedimiento de ramificación y acotación en el ejemplo 7.2.

Paso 6. (control de optimalidad). Puesto que la lista de problemas a procesar no está vacía, se continúa con el paso 3.

Paso 3. (solución). El problema P_4 es factible. Su solución es:

$$x_1 = 0; \;\; x_2 = 1; \;\; x_3 = 4.5; \;\; x_4 = 0.5; \;\; Z = 2$$

Paso 4. (actualización de cotas). Nada tiene lugar en este paso.

Paso 5. (poda). Puesto que la solución obtenida no satisface las condiciones de integralidad ($x_3 \notin \mathbb{N}$), y el valor correspondiente de la función objetivo es mayor que el valor actual de la cota superior, no podrá lograrse una mejor solución que la disponible llevando a cabo ramificaciones adicionales sobre la rama correspondiente. La rama se poda y se continúa con el paso 6.

Paso 6. (comprobación de optimalidad). Puesto que la lista de problemas a procesar está vacía, y hay una solución candidata a minimizador, esta solución es el minimizador del problema original. La solución óptima es por tanto

$$x_1^* = 0.5 \;\; x_2^* = 0 \;\; x_3^* = 2 \;\; x_4^* = 0.5 \;\; Z^* = 1.5$$

El proceso en su totalidad se ilustra en la figura 7.4.

Obsérvese que en este ejemplo sencillo, se ha empleado una estrategia de procesamiento en anchura.

■

7.2.4 Otros problemas de programación lineal entera-mixta

Otros dos tipos de PPLEMs (más restringidos) tienen interés: (i) problemas de programación lineal entera estricta (PPLEEs) y (ii) problemas de programación lineal entera-mixta 0/1 (PPLEM 0/1).

Los PPLEEs se resuelven mediante la técnica del RA de forma similar a los PPLEMs. En estos problemas todas las variables son candidatas potenciales para ramificar. Los PPLEM 0/1 también se resuelven de forma análoga; en estos problemas ramificar es equivalente a fijar las variables binarias bien a 0, bien a 1.

7.3 El método de los cortes de Gomory

7.3.1 Introducción

Un método alternativo de solución de PPLEM es el procedimiento de los cortes de Gomory. En esta técnica, se resuelve el problema original relajado en el que se incluyen restricciones adicionales, que reducen la región factible sin excluir soluciones que cumplen las condiciones de optimalidad. En cada iteración se añade una restricción que se denomina corte de Gomory. Este procedimiento genera progresivamente una envoltura convexa de la región factible entera-mixta, lo que origina soluciones que cumplen las condiciones de integralidad. Más información sobre este procedimiento se puede encontrar en los manuales de Rao [92], Bronson [17], y en el artículo original de Gomory [50].

A continuación se explica el algoritmo de Gomory para PPLEE. Su generalización para PPLEM se deja como ejercicio al lector.

7.3.2 Generación de un corte

La región factible del problema original es $\mathbf{Ax} = \mathbf{b}$, que empleando la partición estándar del simplex se escribe como

$$\begin{pmatrix} \mathbf{B} & \mathbf{N} \end{pmatrix} \begin{pmatrix} \mathbf{x}_B \\ \mathbf{x}_N \end{pmatrix} = \mathbf{b}$$

o

$$\mathbf{B}\mathbf{x}_B + \mathbf{N}\mathbf{x}_N = \mathbf{b}$$

Despejando $\mathbf{x}_B$ se obtiene

$$\mathbf{x}_B - \mathbf{B}^{-1}\mathbf{N}\mathbf{x}_N = \mathbf{B}^{-1}\mathbf{b}$$

que en forma matricial es

$$\begin{pmatrix} \mathbf{I} & \mathbf{B}^{-1}\mathbf{N} \end{pmatrix} \begin{pmatrix} \mathbf{x}_B \\ \mathbf{x}_N \end{pmatrix} = \mathbf{B}^{-1}\mathbf{b},$$

y empleando la notación estándar del simplex es

$$\begin{pmatrix} \mathbf{I} & \mathbf{U} \end{pmatrix} \begin{pmatrix} \mathbf{x}_B \\ \mathbf{x}_N \end{pmatrix} = \tilde{\mathbf{b}}$$

o

$$\mathbf{x}_B + \mathbf{U}\mathbf{x}_N = \tilde{\mathbf{b}}$$

Sea x_{B_i} una variable básica que ha de ser entera pero no lo es. La fila correspondiente a esta variable en la ecuación anterior es

$$x_{B_i} + \sum_j u_{ij} x_{N_j} = \tilde{b}_i \tag{7.1}$$

Puesto que la solución obtenida es básica y factible, la variables x_{N_j}, que son no básicas, son cero; por tanto, $\tilde{\mathbf{b}} = \mathbf{x}_B$, y puesto que x_{B_i} es no entera, $\tilde{b}_i$ ha de ser no entera.

Por otra parte, cada elemento u_{ij} se puede expresar como suma de una parte entera (positiva, negativa o nula) y una fracción no negativa menor que 1, f_{ij}:

$$u_{ij} = i_{ij} + f_{ij}; \quad \forall j \tag{7.2}$$

De forma análoga, $\tilde{b}_i$ se puede descomponer como

$$\tilde{b}_i = \tilde{i}_i + \tilde{f}_i \tag{7.3}$$

donde $\tilde{i}_i$ es un entero (positivo o negativo) y $\tilde{f}_i$ una fracción no negativa menor que 1. Obsérvese que algunos f_{ij} pueden ser cero, pero $\tilde{f}_i$ ha de ser positivo.

Sustituyendo (7.2) y (7.3) en (7.1) se obtiene

$$x_{B_i} + \sum_j (i_{ij} + f_{ij})\, x_{N_j} = \tilde{i}_i + \tilde{f}_i$$

o

$$x_{B_i} + \sum_j i_{ij} x_{N_j} - \tilde{i}_i = \tilde{f}_i - \sum_j f_{ij} x_{N_j}$$

En la ecuación anterior, el término de la izquierda ha de ser entero puesto que todas sus variables son enteras. Por tanto, el término de la derecha también será entero.

Por otra parte, f_{ij} para todo j, y x_{N_j} para todo j son no negativos, por tanto $\sum_j f_{ij} x_{N_j}$ es no negativo.

El término de la derecha de la ecuación precedente, $\tilde{f}_i - \sum_j f_{ij} x_{N_j}$, es a la vez

- entero, y
- menor que una fracción positiva, $\tilde{f}_i$, menor que 1

y puesto que los enteros que satisfacen esa condición son $0, -1, -2, \ldots$, entonces $\tilde{f}_i - \sum_j f_{ij} x_{N_j}$ es un entero no positivo. Por tanto

$$\tilde{f}_i - \sum_j f_{ij} x_{N_j} \leq 0 \tag{7.4}$$

o

$$\sum_j f_{ij} x_{N_j} - \tilde{f}_i \geq 0 \tag{7.5}$$

Esta última desigualdad se denomina corte de Gomory asociado a la variable básica x_{B_i} que ha de ser entera pero no lo es.

7.3.3 Algoritmo de los cortes de Gomory para PPLE

Entrada. Un PPLE que se quiere resolver.

Salida. Su solución o un mensaje informando de infactibilidad o solución no acotada.

Paso 1. (iniciación). Se resuelve el problema original sin restricciones de integralidad. Si la solución no está acotada o el problema es infactible, se para; el problema original o no está acotado o es infactible.

Paso 2. (control de optimalidad). Si la solución obtenida cumple las condiciones de integralidad, se para dado que esta solución es óptima. En otro caso, se continúa con el paso siguiente.

Paso 3. (generación de corte). Se emplea una variable básica que ha de ser entera pero no lo es para generar un corte de Gomory.

Paso 4. (resolución). Se añade el corte de Gomory obtenido al problema previamente resuelto, se resuelve el problema resultante y se continúa con el paso 2.

Obsérvese que el número de restricciones crece con el número de iteraciones. Puesto que en cada iteración se añade una nueva restricción, debe emplearse para la resolución del problema el MPE (método simplex dual). Esto es así puesto que al añadir una nueva restricción, el problema primal correspondiente se hace infactible pero su dual permanece factible, aunque no óptimo. Por tanto, para resolver el nuevo problema puede partirse de la solución dual del problema previo (sin restricción adicional), lo que supone una ventaja computacional importante. El ejemplo siguiente clarifica lo anterior.

En el problema primal se minimiza

$$Z = \begin{pmatrix} -3 & -5 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

sujeto a

$$\begin{pmatrix} -1 & 0 \\ 0 & -1 \\ -3 & -2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \geq \begin{pmatrix} -4 \\ -6 \\ -18 \end{pmatrix}$$

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \geq \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

Su minimizador es

$$\begin{pmatrix} x_1^* \\ x_2^* \end{pmatrix} = \begin{pmatrix} 2 \\ 6 \end{pmatrix}$$

y el valor óptimo de la función objetivo es $z^* = -36$.

En el problema dual del anterior, se maximiza

$$Z = \begin{pmatrix} y_1 & y_2 & y_3 \end{pmatrix} \begin{pmatrix} -4 \\ -6 \\ -18 \end{pmatrix}$$

sujeto a

$$\begin{pmatrix} y_1 & y_2 & y_3 \end{pmatrix} \begin{pmatrix} -1 & 0 \\ 0 & -1 \\ -3 & -2 \end{pmatrix} \leq \begin{pmatrix} -3 & -5 \end{pmatrix}$$

$$\begin{pmatrix} y_1 & y_2 & y_3 \end{pmatrix} \geq \begin{pmatrix} 0 & 0 & 0 \end{pmatrix}$$

El maximizador es

$$\begin{pmatrix} y_1^* \\ y_2^* \\ y_3^* \end{pmatrix} = \begin{pmatrix} 0 \\ 3 \\ 1 \end{pmatrix}$$

y el valor óptimo de la función objetivo es $z^* = -36$.

Si la restricción adicional $x_2 \leq 4$ se añade al problema primal, éste se convierte en minimizar

$$\begin{pmatrix} -3 & -5 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

sujeto a

$$\begin{pmatrix} -1 & 0 \\ 0 & -1 \\ -3 & -2 \\ 0 & -1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \geq \begin{pmatrix} -4 \\ -6 \\ -18 \\ -4 \end{pmatrix}$$

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \geq \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

Obsérvese que la solución óptima del problema primal original

$$\begin{pmatrix} x_1^* \\ x_2^* \end{pmatrix} = \begin{pmatrix} 2 \\ 6 \end{pmatrix}$$

es infactible para el problema previo.

El problema dual del anterior (con la restricción adicional) consiste en maximizar

$$Z = \begin{pmatrix} y_1 & y_2 & y_3 & y_4 \end{pmatrix} \begin{pmatrix} -4 \\ -6 \\ -18 \\ -4 \end{pmatrix}$$

sujeto a

$$\begin{pmatrix} y_1 & y_2 & y_3 & y_4 \end{pmatrix} \begin{pmatrix} -1 & 0 \\ 0 & -1 \\ -3 & -2 \\ 0 & -1 \end{pmatrix} \leq \begin{pmatrix} -3 & -5 \end{pmatrix}$$

$$\begin{pmatrix} y_1 & y_2 & y_3 & y_4 \end{pmatrix} \geq \begin{pmatrix} 0 & 0 & 0 & 0 \end{pmatrix}$$

Obsérvese que la solución del problema dual previo es factible aunque no óptima para este segundo problema dual

$$\begin{pmatrix} y_1^* \\ y_2^* \\ y_3^* \end{pmatrix} = \begin{pmatrix} 0 \\ 3 \\ 1 \end{pmatrix}$$

con

$$y_4 = 0$$

es una solución factible, pero no óptima, del problema anterior.

Ejemplo 7.3 (algoritmo de los cortes de Gomory). El ejemplo siguiente ilustra el algoritmo de los cortes de Gomory.

Considérese al siguiente PPLEM. Maximizar

$$Z = 120x_1 + 80x_2$$

sujeto a

$$\begin{array}{rrcl} 2x_1 & +x_2 & \leq & 6 \\ 7x_1 & +8x_2 & \leq & 28 \\ x_1 & & \geq & 0 \\ & x_2 & \geq & 0 \\ x_1 & & \in & \mathbb{N} \end{array}$$

cuya región factible se muestra sombreada en la figura 7.5.

El problema anterior, relajado, consiste en maximizar

$$Z = 120x_1 + 80x_2$$

sujeto a

$$\begin{array}{rrrrcl} 2x_1 & +x_2 & +x_3 & & = & 6 \\ 7x_1 & +8x_2 & & +x_4 & = & 28 \\ & & \multicolumn{2}{r}{x_1, x_2, x_3, x_4} & \geq & 0 \end{array}$$

La solución de este problema es (véase el punto P_1 en la figura 7.5)

$$z^* = 391.11$$

y

$$\tilde{\mathbf{b}} = \mathbf{x}_B^* = \begin{pmatrix} x_1^* \\ x_2^* \end{pmatrix} = \begin{pmatrix} \frac{20}{9} \\ \\ \frac{14}{9} \end{pmatrix}, \quad (7.6)$$

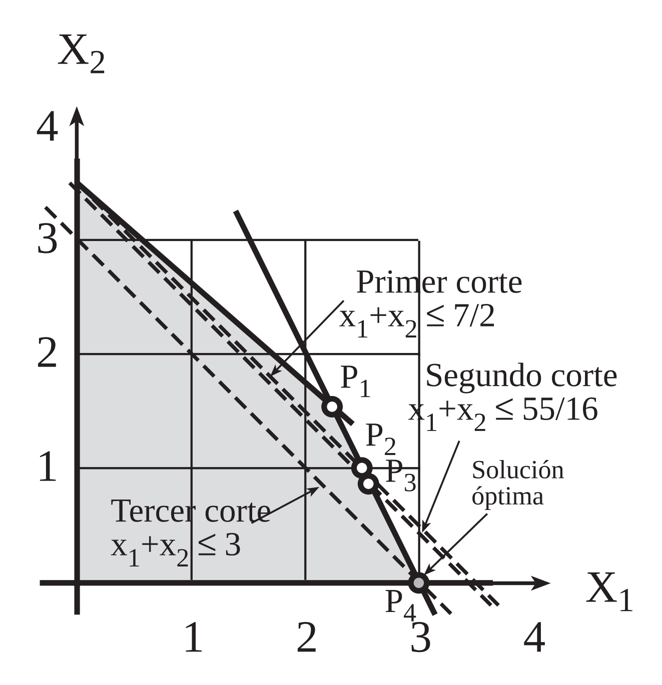


Figura 7.5: Ilustración gráfica de los cortes de Gomory en el Ejemplo 7.3.

$$\mathbf{U}^* = \begin{pmatrix} \frac{8}{9} & -\frac{1}{9} \\ -\frac{7}{9} & \frac{2}{9} \end{pmatrix}. \tag{7.7}$$

Si se emplea la variable x_2 para general un corte de Gomory, la ecuación (7.1) se escribe como $x_2 - \frac{7}{9}x_3 + \frac{2}{9}x_4 = \frac{14}{9}$. Por tanto, se obtiene:

$$\begin{array}{rclcrclcrcl} u_{21} & = & i_{21} + f_{21} & \Leftrightarrow & -\frac{7}{9} & = & -1 + \frac{2}{9} & \Rightarrow & f_{21} & = & \frac{2}{9} \\ u_{22} & = & i_{22} + f_{22} & \Leftrightarrow & \frac{2}{9} & = & 0 + \frac{2}{9} & \Rightarrow & f_{22} & = & \frac{2}{9} \\ \tilde{b}_2 & = & \tilde{i}_2 + \tilde{f}_2 & \Leftrightarrow & \frac{14}{9} & = & 1 + \frac{5}{9} & \Rightarrow & \tilde{f}_2 & = & \frac{5}{9} \end{array}$$

y el corte (7.4) es $\frac{5}{9} - \left(\frac{2}{9} \quad \frac{2}{9} \right) \begin{pmatrix} x_3 \\ x_4 \end{pmatrix} \leq 0$ ó $x_3 + x_4 \geq \frac{5}{2}$.

Nótese que este corte puede expresarse en función de las variables originales x_1 y x_2, de la forma siguiente. Empleando las restricciones de igualdad del problema relajado en forma estándar, x_3 y x_4 pueden expresarse en función de x_1 y x_2:

$$\begin{array}{rcl} x_3 & = & 6 - 2x_1 - x_2, \\ x_4 & = & 28 - 7x_1 - 8x_2 \end{array}$$

para obtener el primer corte (véase la línea a trazos en la figura 7.5): $x_1+x_2 \leq \frac{7}{2}$. Si esta restricción se representa en $\mathbb{R}^2$ sobre la región factible del problema original (expresado en forma canónica), se observa una reducción efectiva de la región factible. Sin embargo, no se excluye ninguna solución entera.

El siguiente problema a resolver es maximizar

$$Z = 120x_1 + 80x_2$$

sujeto a

$$\begin{array}{rrrrrcl} 2x_1 & +x_2 & +x_3 & & & = & 6 \\ 7x_1 & +8x_2 & & +x_4 & & = & 28 \\ & & x_3 & +x_4 & -x_5 & = & \dfrac{5}{2} \\ & & & & x_1, x_2, x_3, x_4, x_5 & \geq & 0 \end{array}$$

Obsérvese que el corte de Gomory se escribe incluyendo una variable artificial x_5.

Su solución es (véase el punto P_2 en la figura 7.5)

$$Z^* = 380; \;\; \mathbf{x}_B^* = \begin{pmatrix} x_1^* \\ x_4^* \\ x_2^* \end{pmatrix} = \begin{pmatrix} \frac{5}{2} \\ \frac{5}{2} \\ 1 \end{pmatrix}; \;\; \mathbf{U}^* = \begin{pmatrix} 1 & -\frac{1}{9} \\ 1 & -1 \\ -1 & \frac{2}{9} \end{pmatrix}$$

Empleando x_1 para generar un nuevo corte, y puesto que $\tilde{b}_1 = x_1^* = \frac{5}{2}$, la ecuación (7.1) es $x_1 + x_3 - \frac{1}{9}x_5 = \frac{5}{2}$.

Por tanto, se puede escribir

$$\begin{array}{rclcrclcrcl} u_{11} & = & i_{11} + f_{11} & \Leftrightarrow & 1 & = & 1+0 & \Rightarrow & f_{11} & = & 0 \\ u_{12} & = & i_{12} + f_{12} & \Leftrightarrow & -\frac{1}{9} & = & -1+\frac{8}{9} & \Rightarrow & f_{21} & = & \frac{8}{9} \\ \tilde{b}_1 & = & \tilde{i}_1 + \tilde{f}_1 & \Leftrightarrow & \frac{5}{2} & = & 2+\frac{1}{2} & \Rightarrow & \tilde{f}_1 & = & \frac{1}{2} \end{array}$$

y el corte (7.4) toma la forma $\frac{1}{2} - (\,0 \quad \frac{8}{9}\,) \begin{pmatrix} x_3 \\ x_5 \end{pmatrix} \leq 0$ ó $x_5 \geq \frac{9}{16}$

Este segundo corte, en función de las variables originales x_1 y x_2 tiene la forma (véase la línea de trazos en la figura 7.5): $x_1 + x_2 \leq \frac{55}{16}$.

El siguiente problema a resolver consiste en maximizar

$$Z = 120x_1 + 80x_2$$

sujeto a

$$\begin{array}{rrrrrrcl} 2x_1 & +x_2 & +x_3 & & & & = & 6 \\ 7x_1 & +8x_2 & & +x_4 & & & = & 28 \\ & & x_3 & +x_4 & -x_5 & & = & \dfrac{5}{2} \\ & & & & x_5 & -x_6 & = & \dfrac{9}{16} \\ & & & & \multicolumn{2}{r}{x_1, x_2, x_3, x_4, x_5, x_6} & \geq & 0 \end{array}$$

La solución es (véase el punto P_3 en la figura 7.5)

$$Z^* = 377.5; \;\; \mathbf{x}_B^* = \begin{pmatrix} x_1^* \\ x_4^* \\ x_5^* \\ x_2^* \end{pmatrix} = \begin{pmatrix} \frac{41}{16} \\ \frac{49}{16} \\ \frac{9}{16} \\ \frac{7}{8} \end{pmatrix}; \;\; \mathbf{U}^* = \begin{pmatrix} 1 & -\frac{1}{9} \\ 1 & -1 \\ 0 & -1 \\ -1 & \frac{2}{9} \end{pmatrix}$$

Empleando x_2 para generar un nuevo corte, y puesto que $\tilde{b}_2 = \frac{7}{8}$, la ecuación (7.1) pasa a ser $x_2 - x_3 + \frac{2}{9}x_6 = \frac{7}{8}$.
Por tanto, se puede escribir

$$\begin{array}{rclcrclcrcl} u_{41} & = & i_{41} + f_{41} & \Leftrightarrow & -1 & = & -1 + 0 & \Rightarrow & f_{41} & = & 0 \\ u_{42} & = & i_{42} + f_{42} & \Leftrightarrow & \frac{2}{9} & = & 0 + \frac{2}{9} & \Rightarrow & f_{42} & = & \frac{2}{9} \\ \tilde{b}_4 & = & \tilde{i}_4 + \tilde{f}_4 & \Leftrightarrow & \frac{7}{8} & = & 0 + \frac{7}{8} & \Rightarrow & \tilde{f}_4 & = & \frac{7}{8} \end{array}$$

y el corte (7.4) pasa a ser $\frac{7}{8} - (\,0 \quad \frac{2}{9}\,) \begin{pmatrix} x_3 \\ x_6 \end{pmatrix} \leq 0$ ó $x_6 \geq \frac{63}{16}$.
En función de las variables originales este tercer corte es

$$x_1 + x_2 \leq 3$$

El siguiente problema a resolver consiste en maximizar

$$Z = 120x_1 + 80x_2$$

sujeto a

$$\begin{array}{rrrrrrrcl}
2x_1 & +x_2 & +x_3 & & & & & = & 6 \\
7x_1 & +8x_2 & & +x_4 & & & & = & 28 \\
 & & x_3 & +x_4 & -x_5 & & & = & \frac{5}{2} \\
 & & & & x_5 & -x_6 & & = & \frac{9}{16} \\
 & & & & & x_6 & -x_7 & = & \frac{63}{16} \\
 & & & & & & x_1,x_2,x_3,x_4,x_5,x_6,x_7 & \geq & 0
\end{array}$$

La solución es (véase el punto P_4 en la figura 7.5):

$$z^* = 360; \;\; \mathbf{x}_B^* = \begin{pmatrix} x_3^* \\ x_4^* \\ x_5^* \\ x_6^* \\ x_1^* \end{pmatrix} = \begin{pmatrix} 0 \\ 7 \\ \frac{9}{2} \\ \frac{63}{16} \\ 3 \end{pmatrix}.$$

Esta solución satisface la condiciones de integralidad y por tanto es el minimizador del problema original. Esta solución es pues

$$x_1^* = 3; \;\; x_2^* = 0; \;\; z^* = 360$$

■

Ejercicios

7.1 Una empresa manufacturera transforma materia prima en dos productos distintos. Esta empresa dispone de 6 unidades de materia prima y 28 horas de tiempo productivo. La manufactura de una unidad del producto I requiere 2 unidades de material y 7 horas de tiempo, mientras que la producción del producto II requiere 1 unidad de material y 8 horas de tiempo. Los precios de venta de los producto I y II son respectivamente \$120 y \$80, respectivamente. Determínese el número de productos de cada tipo que ha de producir la empresa para maximizar su beneficio. Resuélvase el problema empleando un algoritmo de RA y el algoritmo de los cortes de Gomory.

7.2 Un empresario tiene dos almacenes de lámparas eléctricas que contienen respectivamente 1200 y 100 lámparas. Este empresario suministra 3 centros comerciales cuyas demandas respectivas son 100, 700, y 500 lámparas. Los costes de transporte se muestran en la tabla siguiente.

	Centro 1	Centro 2	Centro 3
Almacén 1	14	13	11
Almacén 2	13	13	12

Determínese el número de lámparas que se deben mandar de cada almacén a cada centro comercial para, suministrando la demanda, el beneficio del empresario sea máximo. Resuélvase este problema mediante el algoritmo del RA y mediante el algoritmo de los cortes de Gomory.

7.3 El mantenimiento predictivo de una determinada planta industrial requiere llevar a cabo 4 tareas sucesivas. La compañía que lleva a cabo este mantenimiento cuenta con 6 trabajadores especializados. El tiempo que necesita cada trabajador para llevar a cabo cada tarea se muestra en la tabla siguiente.

	Tarea 1	Tarea 2	Tarea 3	Tarea 4
Trabajador 1	65	73	63	57
Trabajador 2	67	70	65	58
Trabajador 3	68	72	69	55
Trabajador 4	67	75	70	59
Trabajador 5	71	69	75	57
Trabajador 6	69	71	66	59

Suponiendo que cada trabajador sólo puede hacerse cargo de una tarea de mantenimiento, formúlese un PPLE para determinar qué trabajador ha de llevar a cabo cada tarea, de tal manera que se minimice el tiempo total que requiere el proceso de mantenimiento. Este problema es un ejemplo del problema general de asignación de tareas.

7.4 Un operario especializado ha de reparar una instalación de alta montaña. Es conveniente que lleve consigo 5 equipos diferentes de reparación. Sin embargo, el peso máximo a transportar está limitado a 60 unidades. El peso de cada equipo y un parámetro que cuantifica su utilidad esperada aparecen en la tabla siguiente.

Equipo	1	2	3	4	5
Peso	52	23	35	15	7
Utilidad (%)	100	60	70	15	15

¿Qué equipos ha de llevarse consigo el operario? Téngase en cuenta que este problema es un ejemplo del problema general de la mochila.

7.5 Un productor de electricidad debe planificar su producción horaria de energía para maximizar sus beneficios por venta de la misma. Considerando un horizonte de planificación de 2 horas, formular y resolver el PPLEM de planificación óptima. Téngase en cuenta lo siguiente:

(a) El productor produce 5 unidades de energía justo antes de las 2 horas de estudio.

(b) Los precios horarios de venta de energía eléctrica son 6 y 2, respectivamente.

(c) Si está funcionando, la producción máxima horaria es 10 y la mínima 2.

(d) Con motivo de limitaciones técnicas en la subida y bajada de carga, la producción horaria en dos horas consecutivas no puede diferenciarse en más de 4 unidades de energía.

(e) El coste de producción de cada unidad de energía es 3.

7.6 Una empresa consumidora de energía eléctrica puede auto-producirla o comprarla en el mercado mayorista de la electricidad. Las demandas eléctricas de la empresa para las dos próximas horas son 8 y 10 respectivamente. Los precios de la energía en el mercado de esas dos horas son 6 y 8 respectivamente. El grupo auto-productor de la empresa, si funciona, ha de hacerlo con una producción entre 2 y 6, y no puede variar su producción en dos horas consecutivas más de 4. El coste de producción de este grupo auto-productor es 7, y está produciendo 7 unidades de energía previamente a las dos horas de análisis. Determínese el nivel óptimo de auto-producción y de compra en el mercado para las dos horas consideradas.

Capítulo 8

Optimalidad y dualidad en programación no lineal

La programación lineal es una de las mayores contribuciones al campo de la toma científica de decisiones. Su versatilidad y adaptabilidad ha hecho que este modelo tenga aplicación en casi todos los campos de la ingeniería y de la ciencia. No obstante, ciertas clases de problemas requieren tener en cuenta los aspectos no lineales del mismo. Esta necesidad ha conducido a investigar tanto los aspectos teóricos como los computacionales de los problemas de programación no lineal. En este capítulo se abordan diversas generalizaciones del modelo lineal que nos lleva al campo de la programación no lineal. En la sección 8.1 se describe el problema de programación no lineal y se introducen algunos conceptos importantes. En la sección 8.2 se introducen las condiciones necesarias de optimalidad, describiendo y discutiendo las importantes condiciones de optimalidad de Karush–Kuhn–Tucker. La sección 8.3 se dedica a justificar que las condiciones de Karush–Kuhn–Tucker son condiciones suficientes de optimalidad para los problemas convexos. La sección 8.4 se dedica a la teoría de dualidad, analizando especialmente el caso de los problemas convexos. Finalmente, en la sección 8.6, se muestran algunas condiciones, denominadas cualificaciones de restricciones, que garantizan que para un problema no lineal determinado, las condiciones de Karush–Kuhn–Tucker son efectivamente condiciones necesarias de optimalidad.

8.1 Introducción

El *problema de programación no lineal* (PPNL), se puede formular como sigue. Minimizar

$$Z = f(x_1, \ldots, x_n)$$

sujeto a

$$\begin{array}{rcl} h_1(x_1,\ldots,x_n) & = & 0 \\ \vdots & \vdots & \vdots \\ h_\ell(x_1,\ldots,x_n) & = & 0 \\ g_1(x_1,\ldots,x_n) & \leq & 0 \\ \vdots & \vdots & \vdots \\ g_m(x_1,\ldots,x_n) & \leq & 0 \end{array}$$

El modelo anterior se puede formular de una forma compacta como: Minimizar

$$Z = f(\mathbf{x})$$

sujeto a

$$\begin{array}{rcl} \mathbf{h}(\mathbf{x}) & = & \mathbf{0} \\ \mathbf{g}(\mathbf{x}) & \leq & \mathbf{0} \end{array}$$

donde $\mathbf{x} = (x_1,\ldots,x_n)^T$ es el vector de las *variables de decisión*, $f : \mathbb{R}^n \to \mathbb{R}$ es la función objetivo, y $\mathbf{h} : \mathbb{R}^n \to \mathbb{R}^\ell$ y $\mathbf{g} : \mathbb{R}^n \to \mathbb{R}^m$, donde $\mathbf{h}(\mathbf{x}) = (h_1(\mathbf{x}),\ldots,h_\ell(\mathbf{x}))^T$ y $\mathbf{g}(\mathbf{x}) = (g_1(\mathbf{x}),\ldots,g_m(\mathbf{x}))^T$ son, respectivamente, las restricciones de *igualdad* y *desigualdad*. Para que este problema sea propiamente no lineal, al menos una de las funciones involucradas en su formulación debe serlo. Cualquier vector $\mathbf{x} \in \mathbb{R}^n$ que satisface las restricciones se denomina *solución factible*, y el conjunto de todas las soluciones factibles se denomina *región factible*.

Los problemas de optimización no lineal son más difíciles de resolver que los problemas lineales. Estas mayores dificultades aparecen incluso en el caso más simple de minimizar una función de una variable en $\mathbb{R}$ (sin restricciones). Este problema se formula como la minimización de

$$Z = f(x)$$

sujeto a

$$x \in \mathbb{R}.$$

El matemático francés Pierre de Fermat trató este problema en su libro *Methodus ad disquirendam maximan et minimam* en el siglo XVII (véase Boyer [14]) mediante el estudio de la posición de las rectas tangentes. La figura 8.1(a) muestra un problema en el que su mínimo se alcanza en un punto del conjunto de puntos donde la recta tangente es horizontal. Existe otro tipo de problemas en los que los mínimos relativos no satisfacen la condición anterior (véase la figura 8.1(b)).

Las funciones anteriores muestran una primera dificultad: la caracterización de los candidatos a mínimo. En un problema general, la función no necesariamente posee recta tangente (o hiperplano tangente en el caso de funciones de

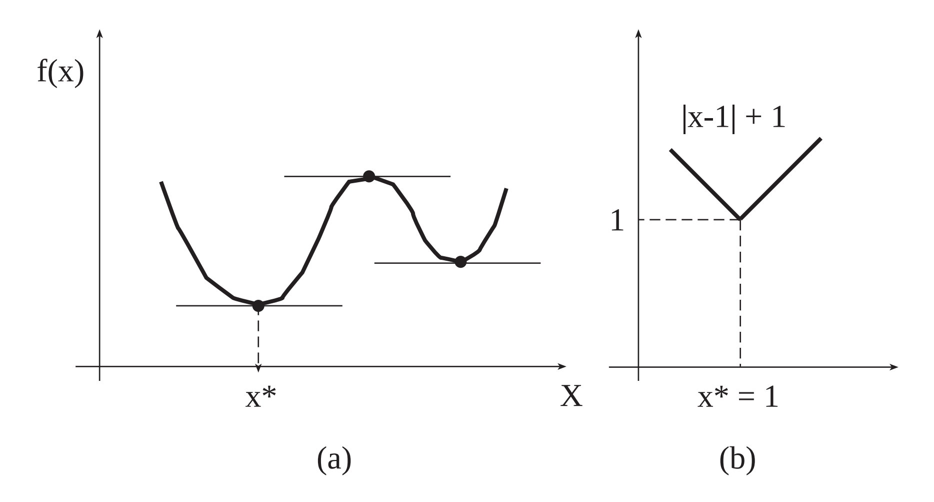


Figura 8.1: Mínimos locales y globales.

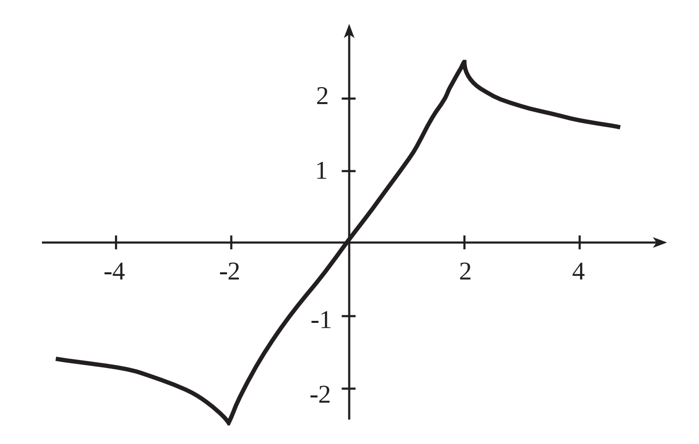


Figura 8.2: Gráfica de la función $f(x) = (x+2)^{2/3} - (x-2)^{2/3}$.

varias variables); para poseer recta tangente la función debe ser *diferenciable.* El ejemplo siguiente ahonda en esta cuestión.

Si se considera el problema de buscar el mínimo de la función (véase la figura 8.2):

$$f(x) = (x+2)^{2/3} - (x-2)^{2/3}$$

en $\mathbb{R}$, se encontraría una dificultad importante. El criterio de buscar puntos donde se anula la primera derivada no conduciría a ningún candidato. Pero por otro lado, el hecho de que la función f tiende a cero cuando x tiende a $\pm\infty$ y que la misma toma valores negativos, indica que f debe alcanzar su mínimo en algún punto. ¿Dónde está la contradicción? Un análisis de f revela que el mínimo es alcanzado en $x = -2$, pero f no es diferenciable en dicho punto. Este simple ejemplo pone de manifiesto que se debe ser extremadamente cuidadoso

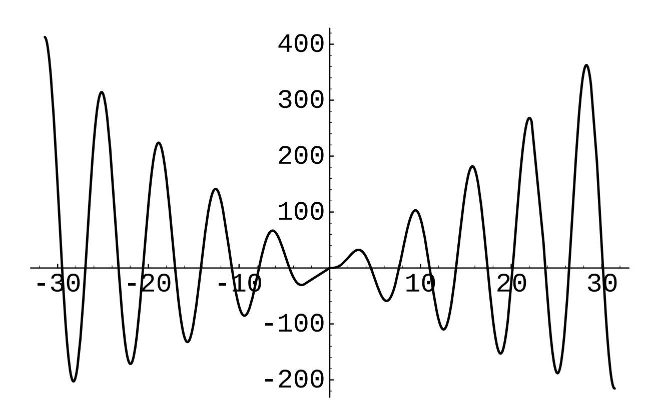


Figura 8.3: Gráfica de la función $f(x) = (1/10)x^2 + 10(\sin x - x\cos x)$.

cuando se está trabajando con funciones que no son diferenciables.

Puesto que este tema escapa de los objetivos perseguidos en este libro, se estudiarán únicamente problemas en los que todas las funciones involucradas sean diferenciables. La teoría de la optimización y los métodos computacionales para estos problemas se denomina *optimización no lineal diferenciable*. Se han desarrollado generalizaciones del concepto de diferenciabilidad para poder abordar problemas más generales de optimización. Las teorías más consistentes en el momento actual se basan en los llamados subgradientes que dan lugar al concepto de subdiferencial (véase Rockafellar [95]) y los gradientes generalizados de Clarke [26]. El lector interesado puede consultar una introducción a estos temas en Jahn [59] y Lemaréchal [65]. El libro de Shimizu et al. [97] recoge, además de los aspectos teóricos, métodos computacionales basados en los subgradientes y en los gradientes generalizados.

La programación no lineal diferenciable no está exenta de dificultades. El hecho de que se puedan caracterizar los candidatos a mínimos de la función no es suficiente para determinarlos explícitamente. Para ilustrarlo, se considerará el siguiente ejemplo (véase la figura 8.3)

$$f(x) = \frac{1}{10}x^2 + 10(\sin x - x\cos x)$$

Esta función es diferenciable en toda la recta real, pero posee una dificultad importante: posee infinitos puntos donde la recta tangente es horizontal, esto es, puntos donde $f'(x) = 0$. Si se quisiera saber cuál de ellos es una solución al problema, se debería evaluar la función objetivo en todos ellos. Este hecho hace inviable, en un tiempo finito, poder abordar numéricamente el problema. Por otro lado, existe la dificultad de resolver de forma cerrada (analíticamente) la ecuación $f'(x) = 0$.

Como conclusión, la existencia de infinitos candidatos además de no poder generarlos explícitamente hace imposible saber, de una forma cierta, si un candidato dado es un mínimo absoluto.

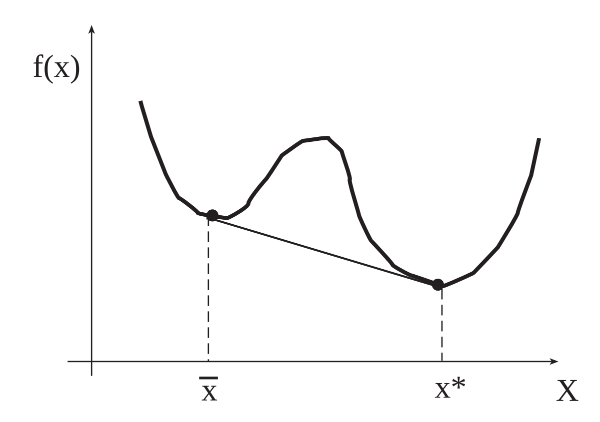


Figura 8.4: Ilustración de la propiedad de convexidad.

Una de las principales diferencias entre la programación lineal y la no lineal es la existencia de los llamados mínimos locales. En ellos se optimiza la función objetivo en un entorno, pero no en toda la región factible.

Esta dificultad motiva una nueva exigencia para la familia de funciones a analizar. Se considerará una clase de funciones en la que sus óptimos locales son también globales. En la figura 8.4 se muestra la gráfica de una función que posee un mínimo local que no es un mínimo global. Se observa que las imágenes de los puntos del intervalo $[\bar{x}, x^*]$ están por encima del segmento que une el mínimo absoluto con el relativo. La condición de convexidad aparece como una condición suficiente que evita este comportamiento. Para que una función sea convexa, su grafo, en cualquier intervalo, debe quedar por debajo del segmento que une las imágenes de los extremos.

Finalmente, es importante señalar que un problema general de programación no lineal puede que no posea solución óptima, esencialmente debido a una de las razones siguientes:

1. La función no está acotada en la región factible S. Por ejemplo, la función de variable real $f(x) = x$, donde $x \in \mathbb{R}$ decrece sin límite a $-\infty$ cuando x tiende a $-\infty$. En este caso se escribirá

$$\text{Infimo}_{\ x \in S} \ \ f(x) = -\infty$$

2. La función está acotada en S, pero ésta no alcanza la mayor de las cotas inferiores en S. Este valor se denota por $\text{ínfimo}_{x \in S} \ \ f(x)$. Por ejemplo, la función $f(x) = e^{-x}$ está acotada en $S = \mathbb{R}$ y la mayor de las cotas inferiores es 0 pero no es alcanzada por $f(x)$.

En muchos casos es importante saber si existe al menos una solución del problema de optimización. Existen condiciones suficientes que lo garantizan, como la proporcionada por el teorema de Weierstrass, que garantiza la existencia de soluciones cuando f es una función continua y el conjunto factible S es cerrado y acotado.

Teorema 8.1 (Weierstrass). *Sea S un conjunto cerrado, acotado y no vacío de $\mathbb{R}^n$ y $f : S \to \mathbb{R}$ una función continua. El problema de minimizar*

$$Z = f(\mathbf{x})$$

sujeto a

$$\mathbf{x} \in S$$

posee al menos una solución óptima. ∎

En ocasiones el conjunto factible no es acotado, y para este caso se puede aplicar el siguiente corolario del teorema anterior.

Corolario 8.1 (existencia de soluciones óptimas.) *Sea S un conjunto cerrado no vacío (posiblemente no acotado) de $\mathbb{R}^n$ y $f : S \to \mathbb{R}$ una función continua. Si*

$$\lim_{\mathbf{x} \to \infty, \mathbf{x} \in S} f(\mathbf{x}) = +\infty$$

entonces el problema de minimizar

$$Z = f(\mathbf{x})$$

sujeto a

$$\mathbf{x} \in S$$

admite solución óptima.

Estos resultados pueden ser más explícitos, como se verá, cuando la función objetivo f sea convexa.

En la siguiente sección se analizarán importantes condiciones necesarias para el PPNL con restricciones, conocidas con el nombre de condiciones de optimalidad de Karush–Kuhn–Tucker. Éstas deben ser satisfechas por todos los mínimos locales de la "mayoría" de los PPNL, en particular por sus mínimos globales. Se motivará cómo aparecen estas condiciones y al final del capítulo, se introducirán las condiciones que garantizan que un determinado PPNL pertenece a esta "mayoría" de problemas para los que son necesarias las condiciones KKT. También se estudiarán hipótesis bajo las cuales las condiciones KKT son suficientes para garantizar que una determinada solución es un mínimo global, lo que nos conducirá al concepto de convexidad, ya enunciado anteriormente. A continuación se tratará la teoría de dualidad en programación matemática, que será de gran importancia en los métodos numéricos analizados en el capítulo 9. Esta teoría de la dualidad se ilustra con un caso de estudio práctico. Finalmente, se establecen de forma breve las condiciones de regularidad necesarias para la formulación de las condiciones de optimalidad anteriores.

8.2 Condiciones necesarias de optimalidad

8.2.1 Diferenciabilidad

En la sección anterior se han mencionado los conceptos de *diferenciabilidad* y *convexidad* para problemas de optimización sin restricciones. En ésta y en la sección siguiente se emplearán estos conceptos en el caso general de problemas de optimización, no necesariamente sin restricciones.

La propiedad de diferenciabilidad permite caracterizar los extremos locales (mínimos o máximos), proporcionando condiciones necesarias para la optimalidad de una solución. Se centrará la atención en la búsqueda de mínimos, ya que los máximos pueden ser obtenidos a partir de la relación

$$\text{Maximizar}_{\mathbf{x}\in S} f(x) = -\text{Minimizar}_{\mathbf{x}\in S} - f(\mathbf{x}).$$

Para una mayor claridad, se da la siguiente definición precisa de este tipo de puntos.

Definición 8.1 (mínimo global). *Una función* $f(\mathbf{x})$ *tiene un mínimo global en el conjunto de puntos* S (*respectivamente, un mínimo global estricto*) *en el punto* $\mathbf{x}^*$, *si y sólo si* $f(\mathbf{x}^*) \leq f(\mathbf{x})$ [*respectivamente,* $f(\mathbf{x}^*) < f(\mathbf{x})$] *para todo* $\mathbf{x}$ *en* S. ■

Definición 8.2 (mínimo local). *Una función* $f(\mathbf{x})$ *tiene un mínimo local* (respectivamente, un mínimo local estricto) *sobre el conjunto* S *en el punto* $\bar{\mathbf{x}}$, *si y sólo si existe un número positivo* ε *cumpliendo* $f(\bar{\mathbf{x}}) \leq f(\mathbf{x})$ (respectivamente, $f(\bar{\mathbf{x}}) < f(\mathbf{x})$) *para todo* $\mathbf{x}$ *en* S *tal que* $0 < \|\bar{\mathbf{x}} - \mathbf{x}\| < \varepsilon$. ■

De estas definiciones se concluye que todo mínimo global también es local.

En una dimensión es fácil ilustrar los conceptos de mínimo local y global (véase la figura 8.5). En este ejemplo, S es el segmento $[a, b]$. $\bar{S} = \{a, x_1, x_2\}$ es el conjunto de óptimos locales y $S^* = \{a, x_2\}$ es el conjunto de mínimos globales.

Se recuerda seguidamente al lector el concepto de diferenciabilidad.

Definición 8.3 (diferenciabilidad). *Se dice que* $f : \mathbb{R}^n \to \mathbb{R}$ *es diferenciable en* $\mathbf{x}$ *si las derivadas parciales* $\partial f/\partial x_i$, $i = 1, \ldots, n$, *existen, y*

$$\lim_{\mathbf{y}\to\mathbf{x}} \frac{f(\mathbf{y}) - f(\mathbf{x}) - \nabla f(\mathbf{x})^T(\mathbf{y} - \mathbf{x})}{\|\mathbf{y} - \mathbf{x}\|} = 0$$

■

Recuérdese que el gradiente de f en $\mathbf{x}$ es el vector definido por

$$\nabla f(\mathbf{x}) = \left(\frac{\partial f(\mathbf{x})}{\partial x_1}, \ldots, \frac{\partial f(\mathbf{x})}{\partial x_n} \right)^T$$

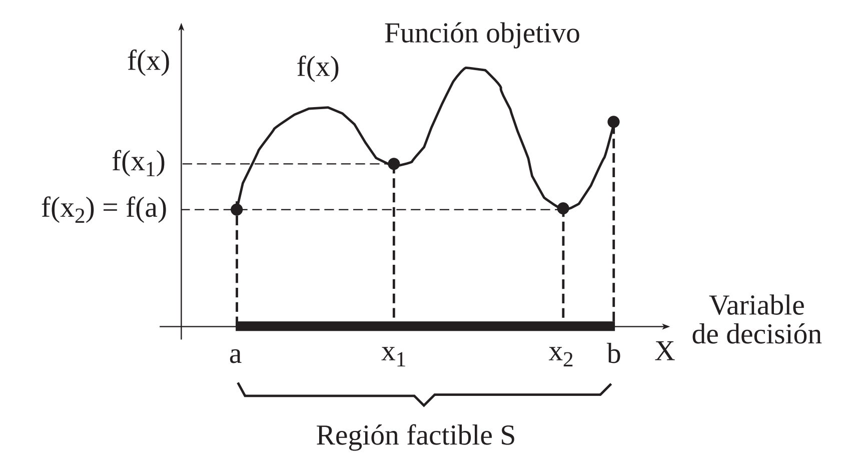


Figura 8.5: Una función con tres mínimos locales y dos globales.

Es bien conocido que el gradiente es un vector que parte de $\mathbf{x}$ es ortogonal a las superficies de nivel (conjunto de puntos que satisfacen $f(\mathbf{x}) = k$ para una constante dada k). El gradiente en el punto $\mathbf{x}$ es la dirección de máximo ascenso para la función f en $\mathbf{x}$.

Definición 8.4 (función continuamente diferenciable). *Una función f se denomina continuamente diferenciable en $\bar{\mathbf{x}}$ si todas sus derivadas parciales son continuas en $\bar{\mathbf{x}}$. En este caso la función también es diferenciable.* ■

En todo este capítulo se considerará el siguiente PPNL. Minimizar

$$Z = f(\mathbf{x}) \tag{8.1}$$

sujeto a

$$\begin{array}{rcl} \mathbf{h}(\mathbf{x}) & = & \mathbf{0} \\ \mathbf{g}(\mathbf{x}) & \leq & \mathbf{0} \end{array} \tag{8.2}$$

donde $f : \mathbb{R}^n \to \mathbb{R}$, $\mathbf{h}:\mathbb{R}^n \to \mathbb{R}^\ell$, $\mathbf{g}:\mathbb{R}^n \to \mathbb{R}^m$ con $\mathbf{h}(\mathbf{x}) = (h_1(\mathbf{x}), \ldots, h_\ell(\mathbf{x}))^T$ y $\mathbf{g}(\mathbf{x}) = (g_1(\mathbf{x}),\ldots, g_m(\mathbf{x}))^T$ son continuamente diferenciables en la región factible $S = \{\mathbf{x}|\mathbf{h}(\mathbf{x}) = \mathbf{0}, \mathbf{g}(\mathbf{x}) \leq \mathbf{0}\}$.

El problema (8.1) sin restricciones ($S = \mathbb{R}^n$) se denomina *problema de optimización no restringido.* Cuando las restricciones (8.2) están presentes, el problema se llama *problema de optimización restringido.*

8.2.2 Condiciones de Karush–Kuhn–Tucker

El resultado teórico más importante en el campo de la programación no lineal es el que lleva a las llamadas condiciones de Karush, Kuhn, y Tucker. Estas condiciones deben ser satisfechas por la solución óptima de cualquier problema lineal y la mayoría de los problemas no lineales. Constituyen la base para el

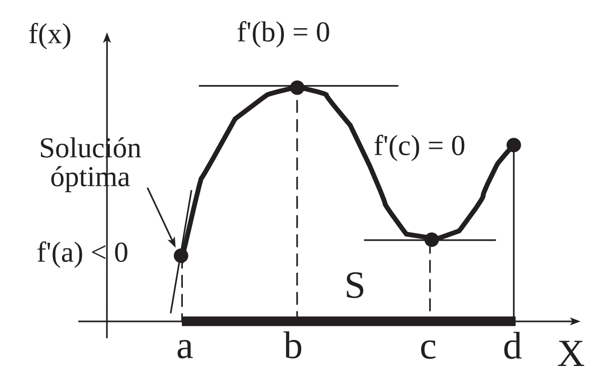


Figura 8.6: En problemas restringidos diferenciables el gradiente no es necesariamente cero en la solución óptima.

desarrollo de muchos algoritmos computacionales y proporcionan un criterio de parada para muchos otros, permitiendo establecer cuando ha sido alcanzado un óptimo local restringido.

En los problemas diferenciables de optimización no restringida la condición necesaria para que una solución sea un mínimo local es que se anule el gradiente. Por el contrario, esta propiedad no es cierta para problemas diferenciables restringidos, como ilustra la figura 8.6 para el punto $\bar{\mathbf{x}} = \mathbf{a}$. Las condiciones de Karush–Kuhn–Tucker generalizan la condición necesaria desarrollada para problemas no restringidos a los problemas con restricciones.

Definición 8.5 (condiciones de Karush–Kuhn–Tucker (CKKT)). *El vector $\bar{\mathbf{x}} \in \mathbb{R}^n$ satisface las CKKT para el PPNL (8.1)–(8.2) si existe un par de vectores $\boldsymbol{\lambda} \in \mathbb{R}^\ell$ y $\boldsymbol{\mu} \in \mathbb{R}^m$ tales que*

$$
\begin{aligned}
\nabla f(\bar{\mathbf{x}}) + \sum_{k=1}^{\ell} \lambda_k \nabla h_k(\bar{\mathbf{x}}) + \sum_{j=1}^{m} \mu_j \nabla g_j(\bar{\mathbf{x}}) &= \mathbf{0} && (8.3)\\
h_k(\bar{\mathbf{x}}) &= 0, k = 1, \ldots, \ell && (8.4)\\
g_j(\bar{\mathbf{x}}) &\leq 0, j = 1, \ldots, m && (8.5)\\
\mu_j g_j(\bar{\mathbf{x}}) &= 0, j = 1, \ldots, m && (8.6)\\
\mu_j &\geq 0, j = 1, \ldots, m && (8.7)
\end{aligned}
$$

■

Los vectores $\boldsymbol{\mu}$ y $\boldsymbol{\lambda}$ se denominan *multiplicadores de Kuhn–Tucker*. La condición (8.6) es conocida con el nombre *condición de complementariedad*, la condición (8.7) requiere la no negatividad de los multiplicadores, y es llamada *condición de factibilidad dual*, y (8.4)–(8.5) se denominan *condiciones de factibilidad primal*.

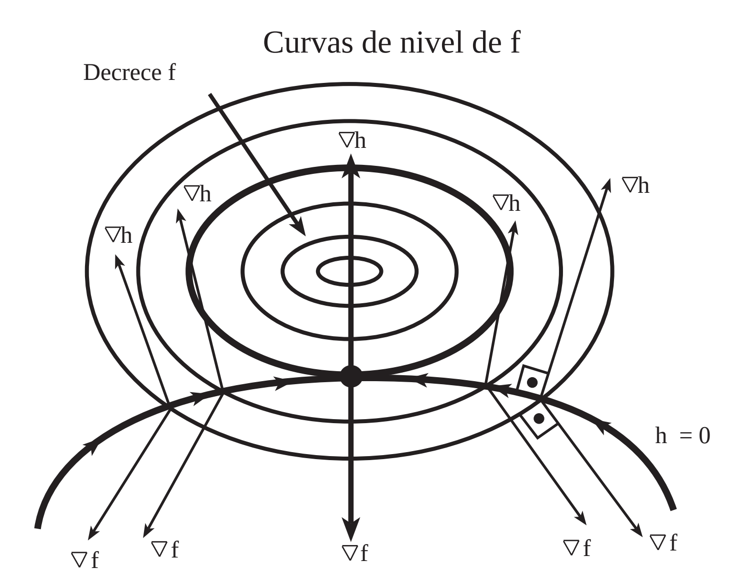


Figura 8.7: Ilustración de las condiciones de Karush–Kuhn–Tucker para el caso de una restricción de igualdad y dos variables.

La génesis de estas condiciones de optimalidad de primer orden (CKKT) puede ser motivada en el caso de dos variables independientes, como se muestra en la figuras 8.7, 8.8, y 8.9.

Considérese el caso de una restricción de igualdad (véase la figura 8.7). Satisfacer la restricción equivale a moverse a lo largo de la curva definida por esta restricción. Las curvas de nivel de la función objetivo son de tal forma que el valor de la función objetivo decrece cuando se lleva a cabo un desplazamiento a lo largo de esta curva según las direcciones de la figura 8.7. La proyección del gradiente de la función objetivo sobre el gradiente de la restricción en un punto dado es negativo. Este hecho indica que dicho desplazamiento disminuye el valor de la función objetivo. Esta proyección se incrementa con el desplazamiento hasta un valor máximo, que corresponde a un punto en el que la curva de nivel de la función objetivo y la de la restricción son tangentes, y por tanto, los gradientes de la función objetivo y de la restricción son paralelos (linealmente dependientes). Si se continuase con el desplazamiento, el valor de la función objetivo se incrementaría, lo que no es conveniente respecto a la perspectiva de minimizar dicha función objetivo. Como conclusión, el mínimo (local) se alcanza en un punto en el que el gradiente de la función objetivo y el de la restricción son linealmente dependientes. Esto es lo que representan las condiciones de optimalidad de primer orden (véase la figura 8.7).

Ahora se considerará el caso de una restricción de desigualdad (véase la figura 8.8), que separa el plano $\mathbb{R}^2$ en dos regiones. En una de ellas la restricción es satisfecha, mientras que en la otra no lo es. La región factible incluye esta región y la curva que define su frontera. Si el mínimo de la función objetivo

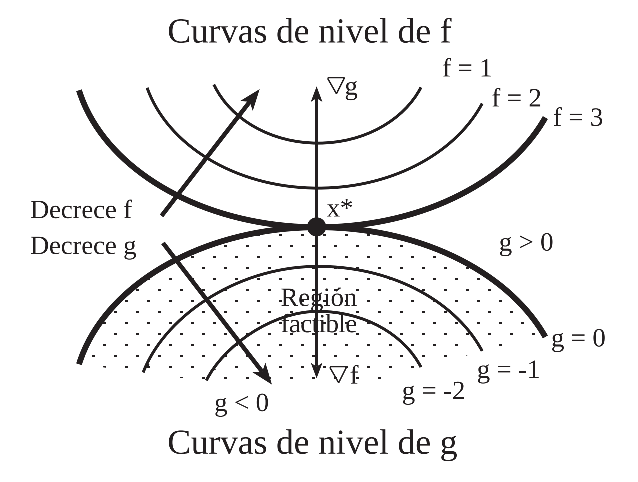


Figura 8.8: Ilustración de las condiciones de Karush–Kuhn–Tucker para el caso de una restricción de igualdad y dos variables.

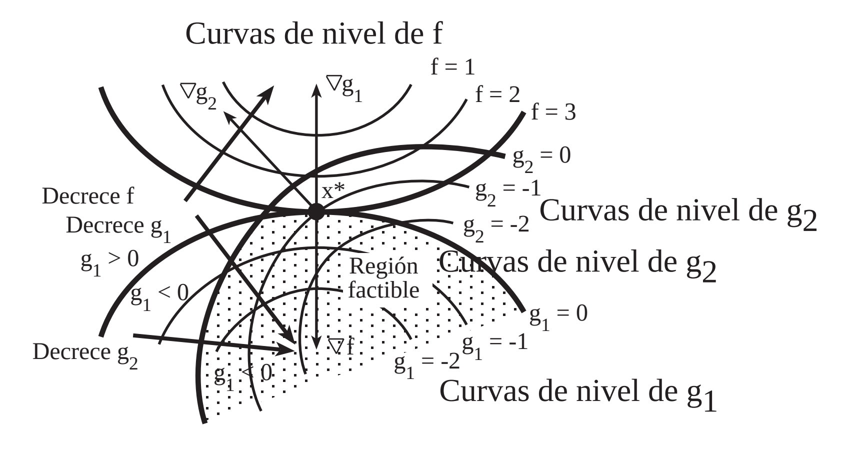


Figura 8.9: Ilustración de las condiciones de Karush–Kuhn–Tucker para el caso de dos restricciones de desigualdad y dos variables.

se alcanza en el interior de la región factible, la restricción no está activa y el correspondiente multiplicador vale cero (véase la figura 8.8). Por el contrario, si el mínimo se alcanza en la frontera, la restricción está activa. En este caso, el problema es equivalente al obtenido al exigir que la restricción se satisfaga con igualdad, y por la discusión anterior, el gradiente de la función objetivo y de la restricción son paralelos en el mínimo. Además, estos vectores tienen sentidos opuestos debido a que la función objetivo aumenta cuando uno se mueve hacia el interior de la región mientras que la restricción tiende a ser negativa y por tanto a disminuir. El multiplicador es entonces positivo o cero. Esto es lo que nos dicen las condiciones de optimalidad en esta situación.

La figura 8.9 ilustra de forma análoga las condiciones KKT para el caso de dos restricciones de desigualdad y dos variables.

Nota 8.1 (casos especiales.) *Si en un determinado PPNL un cierto tipo de restricción (de igualdad o desigualdad) no está presente, estas restricciones y sus respectivos multiplicadores tampoco lo estarán en las correspondientes CKKT. La forma de las CKKT para estos casos son:*

1. **Problemas no restringidos**. *En este caso se tiene sólo la condición*

$$\nabla f(\bar{\mathbf{x}}) = \mathbf{0}.$$

2. **Problemas con sólo restricciones de igualdad**. *Las CKKT son una extensión del método clásico de los multiplicadores. Este método aparece cuando el PPNL tiene exclusivamente restricciones de igualdad, y las CKKT tienen la forma*

$$\begin{aligned} \nabla f(\bar{\mathbf{x}}) + \sum_{k=1}^{\ell} \lambda_k \nabla h_k(\bar{\mathbf{x}}) &= \mathbf{0} \\ h_k(\bar{\mathbf{x}}) &= 0, \; k = 1, \ldots, \ell \end{aligned} \tag{8.8}$$

3. **Problemas con sólo restricciones de desigualdad**. *Las CKKTs son*

$$\begin{aligned} \nabla f(\bar{\mathbf{x}}) + \sum_{j=1}^{m} \mu_j \nabla g_j(\bar{\mathbf{x}}) &= \mathbf{0} \\ g_j(\bar{\mathbf{x}}) &\le 0, \; j = 1, \ldots, m \\ \mu_j g_j(\bar{\mathbf{x}}) &= 0, \; j = 1, \ldots, m \\ \mu_j &\ge 0, \; j = 1, \ldots, m \end{aligned} \tag{8.9}$$

■

Si se define la función lagrangiana

$$\mathcal{L}(\mathbf{x}, \boldsymbol{\mu}, \boldsymbol{\lambda}) = f(\mathbf{x}) + \boldsymbol{\lambda}^T \mathbf{h}(\mathbf{x}) + \boldsymbol{\mu}^T \mathbf{g}(\mathbf{x})$$

se puede escribir las CKKTs como

$$\begin{aligned} \nabla_{\mathbf{x}} \mathcal{L}(\bar{\mathbf{x}}, \boldsymbol{\mu}, \boldsymbol{\lambda}) &= \mathbf{0} \\ \nabla_{\boldsymbol{\lambda}} \mathcal{L}(\bar{\mathbf{x}}, \boldsymbol{\mu}, \boldsymbol{\lambda}) &= \mathbf{0} \\ \nabla_{\boldsymbol{\mu}} \mathcal{L}(\bar{\mathbf{x}}, \boldsymbol{\mu}, \boldsymbol{\lambda}) &\le \mathbf{0} \\ \boldsymbol{\mu}^T \nabla_{\boldsymbol{\mu}} \mathcal{L}(\bar{\mathbf{x}}, \boldsymbol{\mu}, \boldsymbol{\lambda}) &= 0 \\ \boldsymbol{\mu} &\ge \mathbf{0} \end{aligned}$$

Nótese que $\boldsymbol{\mu}^T \nabla_{\boldsymbol{\mu}} \mathcal{L}(\bar{\mathbf{x}}, \boldsymbol{\mu}, \boldsymbol{\lambda}) = 0$ es equivalente a $\mu_j g_j(\bar{\mathbf{x}}) = 0, j = 1, \ldots, m$ debido a que $\nabla_{\boldsymbol{\mu}} \mathcal{L}(\bar{\mathbf{x}}, \boldsymbol{\mu}, \boldsymbol{\lambda}) \le 0$ y $\mu \ge \mathbf{0}$.

A continuación se analiza el caso especial de problemas con restricciones de igualdad.

Considérese el problema: Minimizar

$$Z = f(\mathbf{x}) \tag{8.10}$$

sujeto a

$$\mathbf{h}(\mathbf{x}) = \mathbf{0} \tag{8.11}$$

Las CKKTs para este problema con restricciones sólo de igualdad dan lugar al sistema de ecuaciones no lineales siguiente

$$\begin{array}{rcl} \nabla_{\mathbf{x}}\mathcal{L}(\overline{\mathbf{x}},\overline{\boldsymbol{\lambda}}) & = & \mathbf{0} \\ \mathbf{h}(\overline{\mathbf{x}}) & = & \mathbf{0} \end{array} \tag{8.12}$$

donde $\mathcal{L}(\mathbf{x},\boldsymbol{\lambda}) = f(\mathbf{x}) + \boldsymbol{\lambda}^T\mathbf{h}(\mathbf{x})$.

El sistema anterior de $n+\ell$ ecuaciones puede ser resuelto por el método de Newton. Si $\mathbf{z}$ denota $(\mathbf{x},\boldsymbol{\lambda})$ y $\mathbf{F}(\mathbf{z})$ denota el sistema (8.12), su desarrollo de Taylor es

$$\mathbf{F}(\mathbf{z}+\boldsymbol{\Delta}\mathbf{z}) \approx \mathbf{F}(\mathbf{z}) + \nabla_{\mathbf{z}}\mathbf{F}(\mathbf{z})\ \boldsymbol{\Delta}\mathbf{z}$$

para $||\boldsymbol{\Delta}\mathbf{z}||$ suficientemente pequeño.

Para obtener $\mathbf{F}(\overline{\mathbf{z}}) = \mathbf{0}$, es conveniente encontrar una dirección $\boldsymbol{\Delta}\mathbf{z}$ cumpliendo $\mathbf{F}(\mathbf{z}+\boldsymbol{\Delta}\mathbf{z}) = \mathbf{0}$. Esta dirección se calcula con la condición

$$\nabla_{\mathbf{z}}\mathbf{F}(\mathbf{z})\boldsymbol{\Delta}\mathbf{z} = -\mathbf{F}(\mathbf{z})$$

donde $\nabla_{\mathbf{z}}\mathbf{F}(\mathbf{z})$ puede ser expresada por

$$\nabla_{\mathbf{z}}\mathbf{F}(\mathbf{z}) = \nabla_{(\mathbf{x},\boldsymbol{\lambda})}\mathbf{F}(\mathbf{x},\boldsymbol{\lambda}) = \begin{pmatrix} \nabla_{\mathbf{xx}}\mathcal{L}(\mathbf{x},\boldsymbol{\lambda}) & \boldsymbol{\nabla}_{\mathbf{x}}^T\mathbf{h}(\mathbf{x}) \\ \boldsymbol{\nabla}_{\mathbf{x}}\mathbf{h}(\mathbf{x}) & \mathbf{0} \end{pmatrix}$$

y donde $\boldsymbol{\nabla}_{\mathbf{x}}\mathbf{h}(\mathbf{x})$ es el Jacobiano de $\mathbf{h}(\mathbf{x})$.

La matriz anterior se denomina matriz KKT del problema (8.10)–(8.11), y el sistema

$$\begin{pmatrix} \nabla_{\mathbf{xx}}\mathcal{L}(\mathbf{x},\boldsymbol{\lambda}) & \boldsymbol{\nabla}_{\mathbf{x}}^T\mathbf{h}(\mathbf{x}) \\ \boldsymbol{\nabla}_{\mathbf{x}}\mathbf{h}(\mathbf{x}) & \mathbf{0} \end{pmatrix} \begin{pmatrix} \boldsymbol{\Delta}\mathbf{x} \\ \boldsymbol{\Delta}\boldsymbol{\lambda} \end{pmatrix} \begin{pmatrix} \boldsymbol{\nabla}_{\mathbf{x}}\mathcal{L}(\mathbf{x},\boldsymbol{\lambda}) \\ \mathbf{h}(\mathbf{x}) \end{pmatrix}$$

se denomina sistema KKT del problema (8.10)–(8.11). La resolución del mismo constituye una iteración de Newton para resolver el sistema (8.12).

A continuación se analizan diversos ejemplos ilustrativos.

Ejemplo 8.1 (CKKTs para restricciones de desigualdad). Considérese el problema de optimización consistente en minimizar

$$x^2 + y^2$$

sujeto a

$$\begin{array}{rcl} xy & \geq & 4 \\ x & \geq & 0 \end{array} \tag{8.13}$$

La función lagrangiana es

$$\mathcal{L}(x, y; \mu, \lambda) = x^2 + y^2 + \mu_1(4 - xy) - \mu_2 x$$

y las CKKTs

$$\frac{\partial \mathcal{L}}{\partial x} = 2x - \mu_1 y - \mu_2 = 0 \tag{8.14}$$

$$\frac{\partial \mathcal{L}}{\partial y} = 2y - \mu_1 x = 0 \tag{8.15}$$

$$xy \geq 4 \tag{8.16}$$

$$x \geq 0 \tag{8.17}$$

$$\mu_1(4 - xy) = 0 \tag{8.18}$$

$$\mu_2 x = 0 \tag{8.19}$$

$$\mu_1 \geq 0 \tag{8.20}$$

$$\mu_2 \geq 0 \tag{8.21}$$

Se tienen cuatro casos:

Caso 1. $\mu_1 = 0; \mu_2 = 0$. En este caso, de (8.14) y (8.15) se obtiene $x = y = 0$, que no satisface (8.16). Por tanto, el punto $(0, 0)$ no es un punto de KKT.

Caso 2. $\mu_1 \neq 0; \mu_2 = 0$. En este caso, de (8.14) y (8.15) se deriva $x^2 = y^2$, es decir, $x = \pm y$, que junto a (8.18) conducen a $y = \pm 2$. La expresión (8.14) junto a (8.17) y a (8.20) implica que x e y deben ser positivas, concluyendo que $(2, 2)$ es un punto de KKT.

Caso 3. $\mu_1 = 0; \mu_2 \neq 0$. En este caso, (8.15) conducen a $y = 0$, y por tanto no se satisface (8.16).

Caso 4. $\mu_1 \neq 0; \mu_2 \neq 0$. En este caso, (8.19) conduce a $x = 0$, y entonces no se cumple (8.16).

El único punto de KKT, $(2, 2)$, es la solución óptima, como se ilustra en la figura 8.10.

∎

Ejemplo 8.2 (CKKT para el caso general 1). Considérese el problema de optimización: Minimizar

$$x^2 + y^2$$

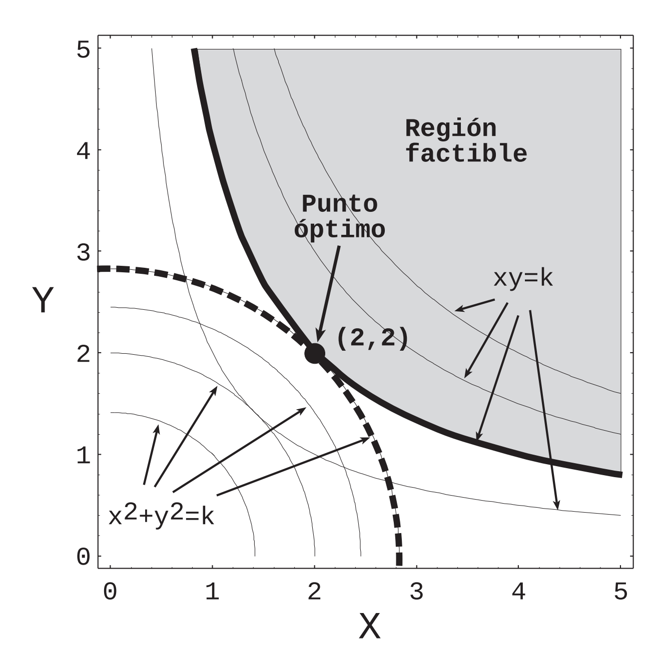


Figura 8.10: Ilustración del problema de minimización del ejemplo 8.1.

sujeto a

$$\begin{aligned} xy &\geq 4 \\ x+y &= 5 \end{aligned} \tag{8.22}$$

La función lagrangiana es

$$\mathcal{L}(x,y;\mu,\lambda) = x^2 + y^2 + \mu(4 - xy) - \lambda(x + y - 5)$$

y las condiciones KKT son

$$\frac{\partial \mathcal{L}}{\partial x} = 2x - \mu y - \lambda = 0 \tag{8.23}$$

$$\frac{\partial \mathcal{L}}{\partial y} = 2y - \mu x - \lambda = 0 \tag{8.24}$$

$$x + y = 5 \tag{8.25}$$

$$xy \geq 4 \tag{8.26}$$

$$\mu(4 - xy) = 0 \tag{8.27}$$

$$\mu \geq 0 \tag{8.28}$$

Se tienen dos casos:

Caso 1. $\mu = 0$. En este caso, de (8.23), (8.24) y (8.25) se obtiene $x = y = \frac{5}{2}$, que satisface (8.26). Entonces, el punto $(\frac{5}{2}, \frac{5}{2})$ es un punto de KKT.

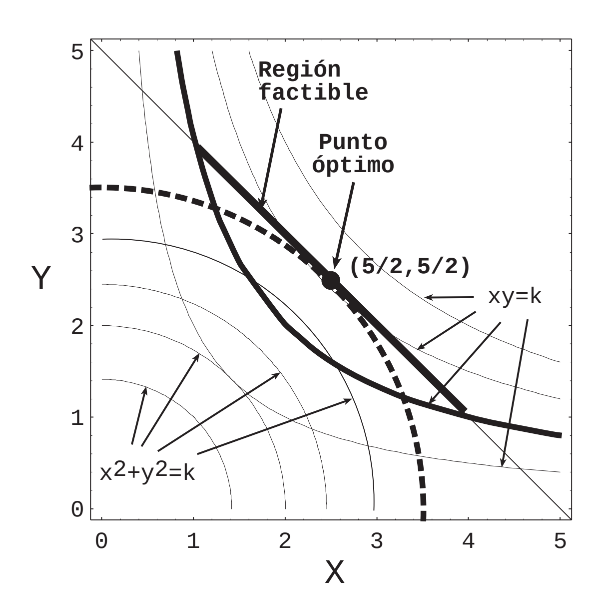


Figura 8.11: Ilustración del problema de minimización del ejemplo 8.2.

Caso 2. $\mu \neq 0$. En este caso, (8.25) y (8.27) dan $x = 4, y = 1$; y $x = 1, y = 4$. Estas dos soluciones conducen, usando (8.23) y (8.24), a $\mu < 0$. Por tanto, se concluye que no son puntos de KKT.

El único punto de KKT, $(\frac{5}{2}, \frac{5}{2})$, es la solución óptima, como se ilustra en la figura 8.11.

■

Ejemplo 8.3 (CKKT para el caso general 2). Considérese el problema de optimización. Minimizar

$$x^2 + y^2 \tag{8.29}$$

sujeto a

$$\begin{array}{rcl} xy & \geq & 4 \\ x + y & = & 5 \\ (x-4)^2 + (y-2)^2 & \leq & 1 \end{array} \tag{8.30}$$

La función lagrangiana es

$$\mathcal{L}(x, y; \mu, \lambda) = x^2 + y^2 + \lambda(x + y - 5) + \mu_1(4 - xy) + \mu_2((x-4)^2 + (y-2)^2 - 1)$$

y las condiciones de KKT son

$$\frac{\partial \mathcal{L}}{\partial x} = 2x + \lambda - \mu_1 y + 2\mu_2(x - 4) = 0 \tag{8.31}$$

$$\frac{\partial \mathcal{L}}{\partial y} = 2y + \lambda - \mu_1 x + 2\mu_2(y - 2) = 0 \tag{8.32}$$

$$x + y = 5 \quad (8.33)$$
$$xy \geq 4 \quad (8.34)$$
$$(x-4)^2 + (y-2)^2 \leq 1 \quad (8.35)$$
$$\mu_1(4 - xy) = 0 \quad (8.36)$$
$$\mu_2((x-4)^2 + (y-2)^2 - 1) = 0 \quad (8.37)$$
$$\mu_1 \geq 0 \quad (8.38)$$
$$\mu_2 \geq 0 \quad (8.39)$$

Se distinguen cuatro casos:

Caso 1. $\mu_1 = 0; \mu_2 = 0$. En este caso, de (8.31)–(8.33) se obtiene que $x = y = \frac{5}{2}$, que no satisface (8.35). Entonces, el punto $(\frac{5}{2}, \frac{5}{2})$ no es de KKT.

Caso 2. $\mu_1 \neq 0; \mu_2 = 0$. En este caso, (8.33) y (8.36) conducen a los puntos candidatos $(4,1)$ y $(1,4)$. El punto $(1,4)$ no satisface (8.35), y $(4,1)$ con (8.31) y (8.32) conducen a $\mu_1 = -2 < 0$, es decir, no es un KKT.

Caso 3. $\mu_1 = 0; \mu_2 \neq 0$. En este caso, (8.33) y (8.35) conducen a $(3,2)$ y $(4,1)$. De (8.31) y (8.32) con $(4,1)$, se obtiene $\mu_1 < 0$; por tanto no es un punto de KKT. Análogamente, de $(3,2)$ se obtiene $\mu_1 = 1 > 0$; por lo que es un punto de KKT.

Caso 4. $\mu_1 \neq 0; \mu_2 \neq 0$. En este caso, (8.33), (8.36) y (8.37) conducen al punto $(4,1)$. Entonces, (8.31) y (8.32) conducen a $\mu_1 = \lambda + 8$ y $\mu_2 = -15 - 3\lambda/2$, que no satisfacen simultáneamente (8.38) y (8.39).

El único punto de KKT, $(3,2)$, es la solución óptima, tal como se ilustra en la figura 8.12.

■

Ejemplo 8.4 (CKKT para el caso general 3). En este ejemplo se calcula los puntos que satisfacen las condiciones de KKT del problema: Minimizar

$$Z = -x_1 + x_2$$

sujeto a

$$\begin{aligned} -x_1^2 + x_2 &= 0 \\ x_1^2 + x_2^2 - 4 &\leq 0 \\ -x_1 &\leq 0 \\ -x_2 &\leq 0 \end{aligned}$$

Se debe introducir un multiplicador por cada restricción. Se denota por μ_1, μ_2, y μ_3 a los multiplicadores de las restricciones de desigualdad, y por λ el multiplicador de la restricción de igualdad.

Las condiciones de KKT se escriben del siguiente modo

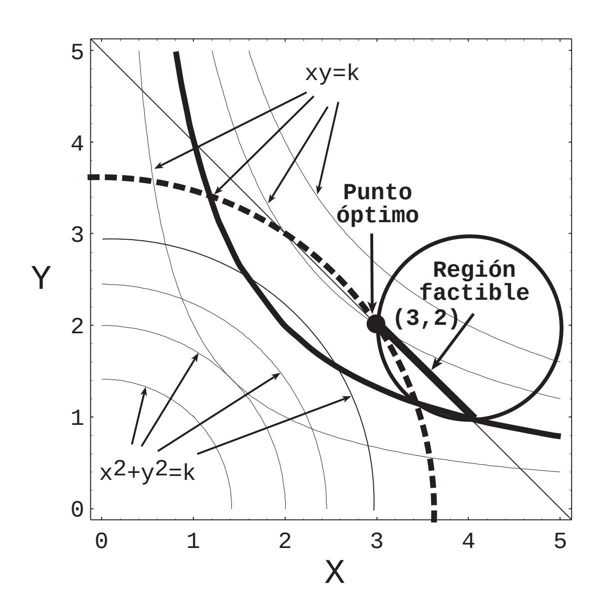


Figura 8.12: Ilustración del problema de minimización del ejemplo 8.3.

1. La condición de estacionariedad de la función lagrangiana es

$$\begin{pmatrix} -1 \\ 1 \end{pmatrix} + \lambda \begin{pmatrix} -2x_1 \\ 1 \end{pmatrix} + \mu_1 \begin{pmatrix} 2x_1 \\ 2x_2 \end{pmatrix} + \mu_2 \begin{pmatrix} -1 \\ 0 \end{pmatrix} + \mu_3 \begin{pmatrix} 0 \\ -1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \tag{8.40}$$

2. La condiciones de factibilidad primal son

$$\begin{aligned} x_1^2 + x_2^2 - 4 &\leq 0 \\ -x_1 &\leq 0 \\ -\, x_2 &\leq 0 \\ -x_1^2 + x_2 &= 0 \end{aligned} \tag{8.41}$$

3. Las condiciones de complementariedad son

$$\mu_1(x_1^2 + x_2^2 - 4) = 0 \tag{8.42}$$

$$\mu_2(-x_1) = 0 \tag{8.43}$$

$$\mu_3(-x_2) = 0 \tag{8.44}$$

4. Las condiciones de factibilidad dual son

$$\mu_1,\ \mu_2,\ \mu_3 \geq 0 \tag{8.45}$$

Para resolver este sistema de igualdades y desigualdades, se consideran los siguientes casos:

Caso 1. $\mu_2 \neq 0$. Si $\mu_2 \neq 0$, las condiciones (8.43), $x_1 = 0$, y (8.40) implican

$$\begin{array}{rcl} -1-\mu_2 & = & 0 \\ 1+2x_2\mu_1-\mu_3+\lambda & = & 0 \end{array}$$

Puesto que $\mu_2 = -1$, no se cumple la no negatividad de los multiplicadores de las desigualdades, por lo que cualquier punto de KKT debe satisfacer $\mu_2 = 0$.

Caso 2. $\mu_3 \neq 0$, y $\mu_2 = 0$. Si $\mu_3 \neq 0$ empleando (8.44), $x_2 = 0$, y usando la relación $-x_1^2 + x_2 = 0$ de (8.41), se obtiene $x_1 = 0$, y de (8.40), se llega a la contradicción

$$-1 = 0$$

Esto muestra que cualquier punto de KKT debe satisfacer $\mu_3 = 0$.

Caso 3. $\mu_1 \neq 0$, y $\mu_2 = \mu_3 = 0$. Si $\mu_1 \neq 0$, de (8.42) se obtiene $x_1^2 + x_2^2 - 4 = 0$, y usando la condición de factibilidad, se llega al siguiente sistema de ecuaciones

$$\begin{array}{rcl} x_1^2+x_2^2-4 & = & 0 \\ -x_1^2+x_2 & = & 0 \end{array} \tag{8.46}$$

La única solución que satisface la condición (8.46) es $\bar{x} = (\sqrt{\delta}, \delta)$, donde $\delta = \dfrac{(-1+\sqrt{17})}{2}$ y, con (8.40), se obtiene

$$\begin{array}{rcl} -1+2\sqrt{\delta}\mu_1-2\sqrt{\delta}\lambda & = & 0 \\ 1+2\delta\mu_1+\lambda & = & 0 \end{array}$$

cuya solución es

$$\begin{array}{rcl} \mu_1 & = & \dfrac{2\delta-\sqrt{\delta}}{2\delta(1+2\delta)} > 0 \\ \lambda & = & \dfrac{-1}{2\sqrt{\delta}} + \dfrac{2\delta-\sqrt{\delta}}{2\delta(1+2\delta)} \end{array}$$

que corresponde a un punto de KKT.

Caso 4. El último caso es $\mu_1 = \mu_2 = \mu_3 = 0$. De (8.40) se obtiene el sistema de ecuaciones

$$\begin{array}{rcl} -1-2\lambda x_1 & = & 0 \\ 1+\lambda & = & 0 \end{array}$$

y se encuentra la solución $\lambda = -1$ y $x_1 = 1/2$. De (8.41), se obtiene $x_2 = x_1^2 = 1/4$. Al ser además un punto factible resulta que también es un punto de KKT. ■

Seguidamente se justifica que las condiciones KKT son necesarias para muchos de los PPNL. Para este fin, se introduce la siguiente definición.

Definición 8.6 (restricción activa). *Dado* $\bar{\mathbf{x}} \in \mathbb{R}^n$*, y dado* $j \in \{1, \ldots, m\}$*. La restricción de desigualdad* $g_j(\mathbf{x}) \leq 0$ *se dice que es activa en el punto* $\bar{\mathbf{x}}$ *si* $g_j(\bar{\mathbf{x}}) = 0$*; se denomina no activa si* $g_j(\bar{\mathbf{x}}) < 0$*. Se denota el conjunto de los índices de las restricciones activas por* $I(\bar{\mathbf{x}})$*:*

$$I(\bar{\mathbf{x}}) = \{j | g_j(\bar{\mathbf{x}}) = 0\}$$

■

La principal motivación para distinguir entre restricciones no activas y activas en un punto es que para las primeras existe un entorno del punto que también satisface la restricción, no siendo así necesariamente para el segundo tipo de restricciones.

Lema 8.1 *Las condiciones de KKT son condiciones necesarias para los óptimos locales de la mayoría de los PPNL.*

Demostración. Se adopta un esquema intuitivo para analizar este punto. Se divide nuestro análisis en dos pasos; primeramente se examina el caso de restricciones de igualdad y posteriormente se aborda el caso general.

1. *Esquema de la demostración de las condiciones de KTT para restricciones de igualdad.* Considérese el PPNL. Minimizar

$$Z = f(\mathbf{x}) \tag{8.47}$$

sujeto a

$$\mathbf{h}(\mathbf{x}) \quad = \quad \mathbf{0} \tag{8.48}$$

donde $f : \mathbb{R}^n \to \mathbb{R}$, $\mathbf{h} : \mathbb{R}^n \to \mathbb{R}^\ell$, $\mathbf{h}(\mathbf{x}) = (h_1(\mathbf{x}), \ldots, h_\ell(\mathbf{x}))^T$ funciones continuamente diferenciables en la región factible $S = \{\mathbf{x} | \mathbf{h}(\mathbf{x}) = \mathbf{0}\}$. Supóngase que $\bar{\mathbf{x}}$ es un mínimo local del problema. Sea

$$\sigma : (-\epsilon, \epsilon) \to \mathbb{R}^n, \;\; \epsilon > 0$$

una curva enteramente contenida en la región factible S tal que

$$\sigma(0) = \bar{\mathbf{x}}, \quad \sigma'(0) = \mathbf{d}$$

Por estar σ contenida en S, se tendrá

$$\mathbf{h}(\sigma(t)) = \mathbf{0}$$

para todo t en el intervalo $(-\epsilon, \epsilon)$. La derivada de esta composición de funciones en 0 debe ser cero porque $\mathbf{h}(\sigma(t))$ es constante, y por la regla de la cadena

$$\nabla \mathbf{h}(\bar{\mathbf{x}})^T \mathbf{d} = 0$$

Por otro lado, la composición $f(\sigma(t))$ debe tener un mínimo local en $t = 0$, y consecuentemente su derivada debe ser cero:

$$\nabla f(\bar{\mathbf{x}})^T \mathbf{d} = 0$$

Además, $\mathbf{d} = \sigma'(0)$ puede ser cualquier vector tangente a la región factible S en $\bar{\mathbf{x}}$. Lo anterior puede escribirse diciendo que si

$$\nabla \mathbf{h}(\bar{\mathbf{x}})^T \mathbf{d} = 0$$

entonces se debe cumplir que

$$\nabla f(\bar{\mathbf{x}})^T \mathbf{d} = 0$$

y esto sólo puede ocurrir si $\nabla f(\bar{\mathbf{x}})$ pertenece al subespacio vectorial generado por los vectores $\nabla \mathbf{h}(\bar{\mathbf{x}})$. Esta dependencia da lugar a los multiplicadores λ_k en (8.3).

2. *Esquema de la demostración de las condiciones de KKT para el caso general.* Minimizar

$$Z = f(\mathbf{x}) \tag{8.49}$$

sujeto a

$$\begin{array}{rcl} \mathbf{h}(\mathbf{x}) & = & \mathbf{0} \\ \mathbf{g}(\mathbf{x}) & \leq & \mathbf{0} \end{array} \tag{8.50}$$

donde $f : \mathbb{R}^n \to \mathbb{R}$, $\mathbf{h}:\mathbb{R}^n \to \mathbb{R}^\ell$, $\mathbf{g}:\mathbb{R}^n \to \mathbb{R}^m$ con $\mathbf{h}(\mathbf{x}) = (h_1(\mathbf{x}), \ldots, h_\ell(\mathbf{x}))^T$ y $\mathbf{g}(\mathbf{x}) = (g_1(\mathbf{x}),\ldots, g_m(\mathbf{x}))^T$ funciones continuamente diferenciables en la región factible $S = \{\mathbf{x}|\mathbf{h}(\mathbf{x}) = \mathbf{0}, \mathbf{g}(\mathbf{x}) \leq \mathbf{0}\}$. La estrategia que se seguirá es analizar esta situación considerando un problema no lineal auxiliar, asociado al mínimo local particular $\bar{\mathbf{x}}$.

Una vez más, sea $\bar{\mathbf{x}}$ un mínimo local de nuestro PPNL. Sea $I(\bar{\mathbf{x}})$ su correspondiente conjunto de restricciones activas. Considérese el problema auxiliar: Minimizar

$$Z = f(\mathbf{x}) \tag{8.51}$$

sujeto a

$$\begin{array}{rcl} g_i(\mathbf{x}) & = & 0, \quad \forall i \in I(\bar{\mathbf{x}}) \\ \mathbf{h}(\mathbf{x}) & = & \mathbf{0} \end{array} \tag{8.52}$$

Es fácil argumentar que el punto $\bar{\mathbf{x}}$ también es un mínimo local del nuevo problema. Entonces, aplicando el primer paso a este problema auxiliar, se concluye la existencia de los multiplicadores $\bar{\mu}$ y $\bar{\lambda}$ donde $\mu_j = 0$ para j no perteneciente a $I(\bar{\mathbf{x}})$ tales que

$$\nabla f(\bar{\mathbf{x}}) + \sum_{k=1}^{\ell} \lambda_k \nabla h_k(\bar{\mathbf{x}}) + \sum_{j=1}^{m} \mu_j \nabla g_j(\bar{\mathbf{x}}) \quad = \quad \mathbf{0} \tag{8.53}$$

El único requisito que falta probar es la no negatividad de los multiplicadores μ_j para j en $I(\bar{\mathbf{x}})$. Para este fin, se fija uno de estos índices en $I(\bar{\mathbf{x}})$, se le llama s y se considera un segundo problema auxiliar definido como sigue. Minimizar

$$Z = f(\mathbf{x}) \tag{8.54}$$

sujeto a

$$\begin{array}{rcl} \mathbf{h}(\mathbf{x}) & = & \mathbf{0} \\ g_i(\mathbf{x}) & = & 0, \quad \forall i \in I(\bar{\mathbf{x}}), i \neq s \\ g_s(\mathbf{x}) & < & 0 \end{array} \tag{8.55}$$

Se elige una curva

$$\sigma : [0, \epsilon) \rightarrow \mathbb{R}^n$$

enteramente contenida en la región factible S de este último problema, tal que

$$\sigma(0) = \bar{\mathbf{x}}, \;\; \sigma'(0) = \mathbf{d}, \;\; \nabla g_s(\bar{\mathbf{x}})^T\mathbf{d} < 0$$

Nótese que esta última exigencia es posible porque la composición

$$g_s(\sigma(t)), \;\; t \in [0, \epsilon)$$

tiene un máximo en $t = 0$, pues $g_s(\sigma(0)) = g_s(\bar{\mathbf{x}}) = 0$ y $g_s(\sigma(t)) \leq 0, \forall t \neq 0$, y su derivada en el punto es precisamente

$$\nabla g_s(\bar{\mathbf{x}})^T\mathbf{d}.$$

Por otro lado, la composición $f(\sigma(t))$ tiene un mínimo en $t = 0$ mientras que

$$g_i(\sigma(t)), i \in I(\bar{\mathbf{x}}), i \neq s$$

son constantes. Consecuentemente

$$\nabla f(\bar{\mathbf{x}})^T\mathbf{d} \geq 0, \;\; \nabla g_i(\bar{\mathbf{x}})^T\mathbf{d} = 0$$

Si se realiza el producto escalar de (8.53) con $\mathbf{d}$, muchos de los términos desaparecen y se obtiene

$$\nabla f(\bar{\mathbf{x}})^T\mathbf{d} + \mu_s \nabla g_s(\bar{\mathbf{x}})^T\mathbf{d} = 0$$

El primer término es no negativo y el factor de μ_s es negativo. Si μ_s fuese también negativo, la igualdad anterior no sería posible. Por tanto, se debe tener

$$\mu_s \geq 0$$

Repitiendo este mismo argumento para cada uno de los índices $s \in I(\bar{\mathbf{x}})$, se obtiene la conclusión buscada sobre el signo de los multiplicadores. ■

Existen algunos detalles técnicos involucrados en las líneas anteriores, y que no se han analizado especialmente en lo relativo a las propiedades específicas que permiten elegir las curvas σ.

Ejemplo 8.5 (óptimo global). En este ejemplo se calculan los óptimos globales (máximo y mínimo) de la función objetivo

$$Z = x_1^3 + x_2^3 + x_3^3$$

sobre la región factible determinada por las siguientes dos restricciones

$$\begin{array}{rcl} x_1^2 + x_2^2 + x_3^2 - 4 & = & 0 \\ x_1 + x_2 + x_3 & \leq & 1 \end{array}$$

Es fácil ver que la región factible es cerrada y acotada, y que la función objetivo es continua, por tanto, la función objetivo alcanza sus valores máximo y mínimo en ese conjunto. Los candidatos a máximo y mínimo se obtienen como solución de las condiciones de KKT. La primera tarea consiste en encontrar todas las soluciones de las condiciones de KKT.

Para este caso particular las condiciones de optimalidad son

$$\begin{array}{rcl}
\begin{pmatrix} 3x_1^2 \\ 3x_2^2 \\ 3x_3^2 \end{pmatrix} + \mu \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} + \lambda \begin{pmatrix} 2x_1 \\ 2x_2 \\ 2x_3 \end{pmatrix} & = & \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \\
x_1^2 + x_2^2 + x_3^2 - 4 & = & 0 \\
\mu(x_1 + x_2 + x_3 - 1) & = & 0 \\
x_1 + x_2 + x_3 & \leq & 1
\end{array}$$

donde μ es el multiplicador de la restricción de desigualdad y λ el de la igualdad. Nótese que los candidatos admisibles de mínimo satisfacen $\mu \geq 0$ y para el máximo serán aquellos puntos en los que se cumplan las condiciones de KKT pero con $\mu \leq 0$.

Para resolver este sistema de igualdades y desigualdades, se considerarán los siguientes casos (enumeración completa de todas las posibilidades):

Caso 1. $\mu = 0$. En este caso el sistema a resolver se simplifica, resultando

$$\begin{array}{rcl}
3x_1^2 + \lambda 2x_1 & = & 0 \\
3x_2^2 + \lambda 2x_2 & = & 0 \\
3x_3^2 + \lambda 2x_3 & = & 0 \\
x_1^2 + x_2^2 + x_3^2 - 4 & = & 0 \\
x_1 + x_2 + x_3 & \leq & 1
\end{array}$$

Debido a que este sistema es invariante respecto a la permutación de las variables, es suficiente estudiar los siguientes casos:

Caso 1a. $x_1 = x_2 = x_3 = 0$. Este caso es imposible debido a la restricción

$$x_1^2 + x_2^2 + x_3^2 - 4 = 0$$

Caso b. $x_1 = x_2 = 0$, $x_3 \neq 0$. En este caso las dos restricciones conducen a la solución única $(0, 0, -2)^T$ y permutando las componentes, se obtiene $(0, -2, 0)^T$ y $(-2, 0, 0)^T$.

Caso 1c. $x_1 = 0$, $x_2, x_3 \neq 0$. La ecuación en la que aparece el multiplicador λ implica $x_2 = x_3$ y las dos restricciones conducen a la solución única $(0, -\sqrt{2}, -\sqrt{2})^T$, y a las obtenidas permutando sus componentes $(-\sqrt{2}, 0, -\sqrt{2})^T$, $(-\sqrt{2}, -\sqrt{2}, 0)^T$.

Caso 1d. $x_1, x_2, x_3 \neq 0$. Las ecuaciones en las que aparece el multiplicador λ conducen a $x_1 = x_2 = x_3$, y las restricciones del problema conducen a la solución única:

$$(-\tfrac{2}{\sqrt{3}}, -\tfrac{2}{\sqrt{3}}, -\tfrac{2}{\sqrt{3}})^T.$$

En principio, todos estos puntos podrían corresponder a máximos o mínimos debido a que el multiplicador μ desaparece (vale $\mu = 0$).

Case 2. $\mu \neq 0$. En este caso el sistema obtenido tiene la forma

$$\begin{aligned}
3x_1^2 + \lambda 2x_1 &= 0 \\
3x_2^2 + \lambda 2x_2 &= 0 \\
3x_3^2 + \lambda 2x_3 &= 0 \\
x_1^2 + x_2^2 + x_3^2 - 4 &= 0 \\
x_1 + x_2 + x_3 &= 1
\end{aligned}$$

Todas las soluciones con $\mu > 0$ serán candidatas a mínimo y las que posean $\mu < 0$ a máximo. Las tres primeras ecuaciones expresan que los vectores $(3x_1^2, 3x_2^2, 3x_3^2)^T$, $(1, 1, 1)^T$, y $(2x_1, 2x_2, 2x_3)^T$ son linealmente dependientes, por lo que el determinante de estos tres vectores debe ser cero. Éste es un determinante de Vandermonde, cuyo valor es $(x_1 - x_2)(x_1 - x_3)(x_2 - x_3)$. Eliminando los multiplicadores de este modo, se resuelve el sistema equivalente

$$\begin{aligned}
(x_1 - x_2)(x_1 - x_3)(x_2 - x_3) &= 0 \\
x_1^2 + x_2^2 + x_3^2 - 4 &= 0 \\
x_1 + x_2 + x_3 &= 1
\end{aligned}$$

Una vez más, por la simetría de la situación, es suficiente considerar el caso particular $x_1 = x_2$. Reemplazando esta ecuación en las otras dos, se consigue

$$\begin{aligned}
x_3 + 2x_1 &= 1 \\
2x_1^2 + (1 - 2x_1)^2 &= 4
\end{aligned}$$

De lo anterior, se obtienen las dos soluciones

$$\frac{1}{6}(2+\sqrt{22},\ \ 2+\sqrt{22},\ \ 2-2\sqrt{22})^T$$

y

$$\frac{1}{6}(2-\sqrt{22},\ \ 2-\sqrt{22},\ \ 2+2\sqrt{22})^T$$

Por otro lado, si se añaden las tres primeras ecuaciones y se consideran las restricciones del problema, se llega a

$$12+3\mu+2\lambda=0$$

que proporciona una relación entre los dos multiplicadores. Si se resuelve en la variable μ y se emplea esta expresión en la primera ecuación, de este caso se consigue

$$3x_1^2+\mu-x_1(12+3\mu)=0$$

y por tanto

$$\mu=\frac{3x_1(4-x_1)}{1-3x_1}$$

Es fácil comprobar, sustituyendo los dos valores obtenidos de x_1, que $\mu<0$ y por tanto el caso II conduce a dos candidatos a máximo.

Una vez más se han identificado los candidatos a máximo y mínimo. Una forma de determinar cuáles de ellos son máximos y cuáles son mínimos consiste en evaluar la función objetivo en ellos y ver dónde se alcanzan los valores extremos. En este ejemplo particular, el mínimo se alcanza en los puntos $(-2,0,0)^T, (0,-2,0)^T, (0,0,-2)^T$ y su valor es -8, y el máximo se alcanza en $\frac{1}{6}(2+\sqrt{22},\ \ 2+\sqrt{22},\ \ 2-2\sqrt{22})^T$ y su valor es $\frac{68-11\sqrt{22}}{18}$. ■

8.3 Condiciones de optimalidad: suficiencia y convexidad

8.3.1 Convexidad

Debido a que la diferenciabilidad es un concepto local (sólo depende de los valores que toma la función en un entorno del punto) esta propiedad permite caracterizar los mínimos locales, sin embargo no es posible emplearla para caracterizar los mínimos globales del PPNL. Por ese motivo, para poder caracterizar los mínimos globales, se añade una nueva exigencia a las funciones que definen el PPNL. La propiedad de *convexidad* de las funciones permite garantizar que todo óptimo local del PPNL también lo es global. Bazaraa et al. [9] analizan otras propiedades globales (que dependen del comportamiento en todo el dominio de definición) que pueden ser aplicadas, estableciendo las relaciones entre éstas y la de convexidad.

Definición 8.7 (función convexa). *Sea $f : S \to \mathbb{R}$, donde S es un conjunto no vacío de $\mathbb{R}^n$. La función f se dice que es convexa en S si para cualquier par de puntos $\mathbf{x}^1$ y $\mathbf{x}^2$, y cualquier escalar λ que cumpla $0 \leq \lambda \leq 1$, se tiene*

$$f\left(\lambda\mathbf{x}^1 + (1-\lambda)\mathbf{x}^2\right) \leq \lambda f(\mathbf{x}^1) + (1-\lambda) f(\mathbf{x}^2). \tag{8.56}$$

Si la desigualdad se satisface estrictamente en (8.56), se dice que f es estrictamente convexa. Similarmente, una función f es cóncava si se cumple la relación (8.56) con la desigualdad inversa, esto es, si la función $(-f)$ es convexa. ■

La figura 8.13 muestra tres ejemplos, uno de una función convexa, otro, de una cóncava; y otro, de una función que no es ni cóncava ni convexa.

Antes de proceder a un análisis más detallado del concepto de convexidad, debe uno convencerse de que esta propiedad garantiza la no existencia de mínimos locales que no sean globales.

Teorema 8.2 (óptimo global y local). *Considérese la función convexa $f : S \to \mathbb{R}$, donde S es un conjunto convexo no vacío de $\mathbb{R}^n$. Todo óptimo local de f en S también es un óptimo global de f en S. Además, si f es estrictamente convexa y posee un mínimo en S, éste es único.* ■

Demostración. Sea $\bar{\mathbf{x}}$ un óptimo local de f en S, entonces existirá un $\epsilon > 0$ cumpliendo

$$f(\bar{\mathbf{x}}) \leq f(\mathbf{x}), \quad \forall \mathbf{x} \in S, \|\mathbf{x} - \bar{\mathbf{x}}\| < \epsilon$$

Para cualquier $\mathbf{y} \in S$, de la convexidad de S se tiene

$$\mathbf{z} = (1-\lambda)\bar{\mathbf{x}} + \lambda\mathbf{y} \in S, \quad \forall \lambda \in [0,1]$$

En particular, para λ suficientemente pequeño, $\|\mathbf{z} - \mathbf{x}\| < \epsilon$, y por la convexidad de f, se obtiene

$$f(\bar{\mathbf{x}}) \leq f(\mathbf{z}) \leq (1-\lambda) f(\bar{\mathbf{x}}) + \lambda f(\mathbf{y})$$

Reorganizando términos, se llega a

$$0 \leq \lambda\left(f(\mathbf{y}) - f(\bar{\mathbf{x}})\right)$$

y por la no negatividad de λ, se concluye que

$$f(\bar{\mathbf{x}}) \leq f(\mathbf{y}).$$

La arbitrariedad de $\mathbf{y} \in S$ implica que $\bar{\mathbf{x}}$ es realmente un óptimo global de f en S.

La unicidad será demostrada por reducción al absurdo. Supóngase que existen dos óptimos diferentes, $\mathbf{x}^1 \neq \mathbf{x}^2$, cumpliendo que $f(\mathbf{x}^1) = f(\mathbf{x}^2)$, y dado un $\lambda \in (0,1)$, por ser f estrictamente convexa, se obtiene

$$f(\lambda\mathbf{x}^1 + (1-\lambda)\mathbf{x}^2) < \lambda f(\mathbf{x}^1) + (1-\lambda) f(\mathbf{x}^2) = f(\mathbf{x}^1) = f(\mathbf{x}^2)$$

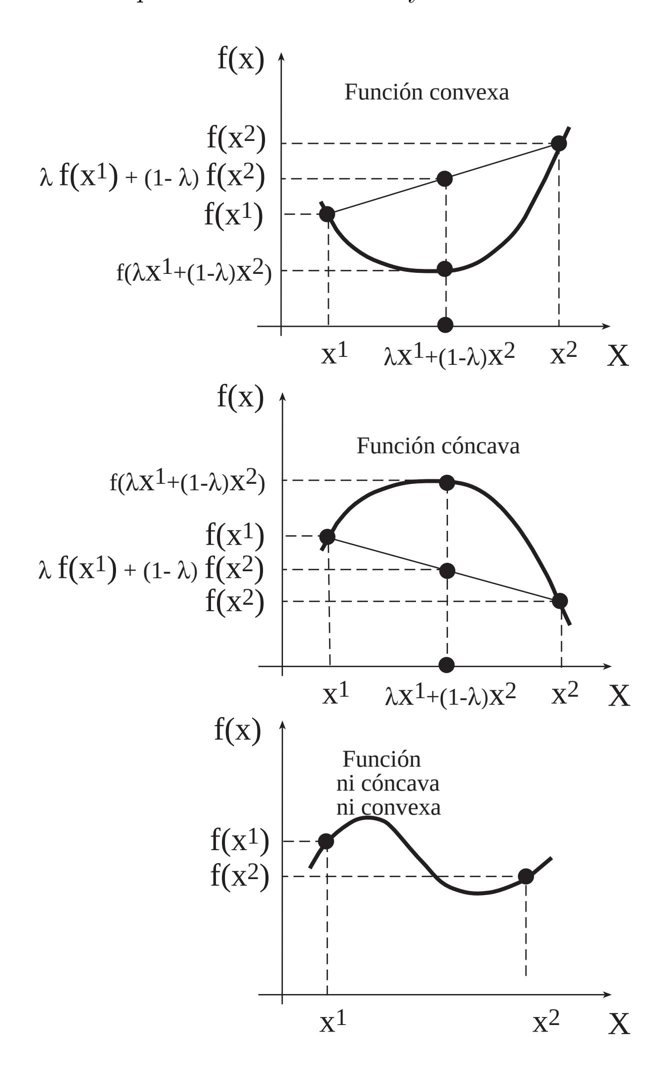


Figura 8.13: Funciones convexa, cóncava y ni cóncava ni convexa.

que contradice la hipótesis de que los valores $f(\mathbf{x}^1)$ y $f(\mathbf{x}^2)$ son los mínimos que toma la función. Esta contradicción garantiza que a lo sumo existe una única solución óptima. ■

Se sabe que la convexidad es una propiedad deseable para las funciones involucradas en el PPNL, especialmente con respecto al tema de los óptimos globales. Se necesita conocer mejor estas funciones, aunque puede ser que el lector ya tenga alguna experiencia con funciones de una variable.

Se comienza considerando la interpretación geométrica de las funciones cóncavas y convexas (véase la figura 8.13). Sean $\mathbf{x}^1$ y $\mathbf{x}^2$ dos puntos distintos de S, y considérese el punto $\lambda\mathbf{x}^1 + (1-\lambda)\mathbf{x}^2 \in S$ para $\lambda \in (0,1)$. Nótese que

$\lambda f(\mathbf{x}^1) + (1-\lambda) f(\mathbf{x}^2)$ da la media ponderada de las imágenes de $f(\mathbf{x}^1)$ y $f(\mathbf{x}^2)$, mientras que $f(\lambda \mathbf{x}^1 + (1-\lambda)\mathbf{x}^2)$ da el valor de f en el punto $\lambda \mathbf{x}^1 + (1-\lambda)\mathbf{x}^2$. Entonces, para una función convexa f, su gráfica, en el intervalo definido por cualquier par de puntos $\mathbf{x}^1$ y $\mathbf{x}^2$ de S, queda por debajo del segmento que une los puntos $[\mathbf{x}^1, f(\mathbf{x}^1)]$ y $[\mathbf{x}^2, f(\mathbf{x}^2)]$. Para funciones cóncavas la gráfica de la función queda por encima del segmento mencionado.

Cuando la función es diferenciable y de una sola variable, la convexidad puede ser caracterizada a través de la derivada de la función. En este caso, la convexidad equivale a que la primera derivada sea una función creciente. Si la función fuese dos veces derivable esta propiedad equivaldría a que la segunda derivada fuese no negativa. Estos criterios, que permiten saber si una función es convexa, también se pueden formular para funciones de varias variables. El papel de la primera derivada lo toma el vector gradiente y el de la segunda derivada, la matriz hessiana, como se pone de manifiesto en los siguientes resultados y definiciones.

Teorema 8.3 (convexidad y diferenciabilidad). *Sea $S \subset \mathbb{R}^n$ un conjunto convexo, y sea $f : \mathbb{R}^n \to \mathbb{R}$ diferenciable en S. Entonces f es una función convexa en S si y sólo si*

$$f(\mathbf{x}^1) - f(\mathbf{x}^2) \geq \nabla f(\mathbf{x}^1)^T (\mathbf{x}^2 - \mathbf{x}^1); \;\; \forall \mathbf{x}^1, \mathbf{x}^2 \in S \tag{8.57}$$

Además, f es estrictamente convexa en S si y sólo si

$$f(\mathbf{x}^1) - f(\mathbf{x}^2) > \nabla f(\mathbf{x}^1)^T (\mathbf{x}^2 - \mathbf{x}^1); \;\; \forall \mathbf{x}^1, \mathbf{x}^2 \in S \text{ tal que } \mathbf{x}^1 \neq \mathbf{x}^2 \tag{8.58}$$

■

Para emplear criterios basados en la segunda derivada, el lector debe recordar los conceptos de funciones dos veces diferenciable y el de matrices semidefinidas positivas.

Definición 8.8 (funciones dos veces diferenciables). *Se dice que $f : \mathbb{R}^n \to \mathbb{R}$ es dos veces diferenciable en el punto $\mathbf{x}$ si existe un vector columna $\nabla f(\mathbf{x})$, y una matriz $n \times n$ $\nabla^2 f(\mathbf{x})$, que cumple*

$$\lim_{\mathbf{y} \to \mathbf{x}} \frac{f(\mathbf{y}) - f(\mathbf{x}) - \nabla f(\mathbf{x})^T (\mathbf{y} - \mathbf{x}) - \frac{1}{2} (\mathbf{y} - \mathbf{x})^T \nabla^2 f(\mathbf{x}) (\mathbf{y} - \mathbf{x})}{\|\mathbf{y} - \mathbf{x}\|^2} = 0.$$

■

La matriz $\nabla^2 f(\mathbf{x})$ se llama *matriz hessiana* de f en el punto $\mathbf{x}$. El elemento ij de $\nabla^2 f(\mathbf{x})$ es la segunda derivada parcial $\partial^2 f(\mathbf{x}) / \partial x_i \partial x_j$. Además, si estas derivadas parciales son continuas entonces f se denomina dos veces continuamente diferenciable y en este caso, $\nabla^2 f(\mathbf{x})$ es una matriz simétrica.

Definición 8.9 (matriz semidefinida positiva). *Una matriz simétrica* $\mathbf{A}$ *es semidefinida positiva si*

$$\mathbf{x}^T\mathbf{A}\mathbf{x} \geq 0$$

para cualquier vector $\mathbf{x}$. *Además, si la igualdad*

$$\mathbf{x}^T\mathbf{A}\mathbf{x} = 0$$

ocurre únicamente en el caso $\mathbf{x} = \mathbf{0}$*, entonces* $\mathbf{A}$ *se dice que es definida positiva.*

■

Equivalentemente, la matriz $\mathbf{A}$ es semidefinida positiva si todos sus valores propios son no negativos, esto es, si todos los valores de λ que cumple la ecuación

$$\det(\mathbf{A} - \lambda\mathbf{I}) = 0$$

son no negativos. La matriz $\mathbf{A}$ es definida positiva cuando todos los valores propios son positivos. Estos valores propios pueden ser calculados por cualquier paquete de cálculo simbólico como el Mathematica, Maple, o Matlab. En el caso de matrices 2×2 un criterio directo para garantizar que la matriz $\mathbf{A} = (a_{ij})$ es definida positiva es que $a_{11} > 0$ y $a_{11}aa_{22} - a_{12}^2 > 0$.

El criterio siguiente caracteriza a las funciones convexas basándose en la matriz hessiana.

Teorema 8.4 (funciones convexas). *Sea* $S \subset \mathbb{R}^n$ *un conjunto abierto y convexo, y sea* $f : \mathbb{R}^n \to \mathbb{R}$ *una función dos veces diferenciable en* S*. Entonces, se cumplen las siguientes propiedades:*

1. f *es una función convexa en* S *si y sólo si* $\nabla^2 f(\tilde{\mathbf{x}})$ *es una matriz semidefinida positiva para todo* $\tilde{\mathbf{x}} \in S$*:*

$$\mathbf{y}^T\nabla^2 f(\tilde{\mathbf{x}})\mathbf{y} \geq 0; \quad \forall \mathbf{y} \in \mathbb{R}^n$$

2. *La función* f *es estrictamente convexa en* S *si y sólo si* $\nabla^2 f(\tilde{\mathbf{x}})$ *es una matriz definida positiva para todo* $\tilde{\mathbf{x}} \in S$*:*

$$\mathbf{y}^T\nabla^2 f(\tilde{\mathbf{x}})\mathbf{y} > 0; \quad \forall \mathbf{y} \in \mathbb{R}^n$$

■

Algunas funciones convexas importantes son:

1. **Funciones afines**: Sea $f(\mathbf{x}) = \mathbf{b}^T\mathbf{x} + a$, donde $a \in \mathbb{R}$ y $\mathbf{b}$, $\mathbf{x} \in \mathbb{R}^n$, una función afín. Entonces, su matriz hessiana es $\nabla^2 f(\tilde{\mathbf{x}}) = 0$ para todo $\tilde{\mathbf{x}} \in \mathbb{R}^n$. Esta función cumple el teorema 8.4 y por tanto es convexa. Nótese que f también es cóncava debido a que $\nabla^2(-f)(\tilde{\mathbf{x}})$ es una matriz de ceros.

2. **Formas cuadráticas**: Sea $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{C}\mathbf{x}+\mathbf{b}^T\mathbf{x}+a$ una forma cuadrática. Si la matriz $\mathbf{C}$ es semidefinida positiva, entonces la función f es convexa, debido a que la matriz hessiana de la forma cuadrática f es $\nabla^2 f(\mathbf{x}) = \mathbf{C}$ para todo $\mathbf{x}$.

En muchas ocasiones se trabaja con combinaciones de funciones convexas. Es por tanto importante saber qué operaciones entre ellas mantienen la propiedad de convexidad. Algunas de estas operaciones son:

1. **Combinación lineal no negativa de funciones convexas**. Sea $f_i(\mathbf{x})$, $i = 1, \ldots, k$ una familia de funciones convexas, y sea λ_i $i = 1, \ldots, k$ un conjunto de escalares positivos. Considérese la función $h(\mathbf{x}) = \sum_{i=1}^{k} \lambda_i f_i(\mathbf{x})$. El hecho de que la suma de matrices semidefinidas positivas sea semidefinida positiva, hace que h sea una función convexa.

2. **Primer tipo de composición de funciones**. Si h es convexa y $\mathbf{T}$ es una función lineal (o afín), la composición $g = h(\mathbf{T})$ también es convexa.

3. **Segundo tipo de composición de funciones**. Si g es convexa, y h una función de una variable convexa no decreciente, la composición $f = h(g)$ es convexa.

4. **Supremo**. El supremo de cualquier familia de funciones convexas es una función convexa.

El lector puede fácilmente justificar estas propiedades.

8.3.2 Condiciones suficientes de Karush–Kuhn–Tucker

Las condiciones de KKT no son suficientes para garantizar que un mínimo local lo sea también global. Por ejemplo, el problema de minimizar

$$Z = x^3$$

sujeto a

$$x \in \mathbb{R}$$

tiene un punto de KKT en $x = 0$, pero no es ni siquiera un mínimo local. Incluso si se supiese que cierto vector corresponde a un mínimo local, no hay forma de garantizar que se trata de un mínimo global.

El próximo lema da una condición necesaria y suficiente de óptimo global para una importante clase de PPNL, en el que la región factible no está explícitamente expresada mediante las funciones de restricción.

Lema 8.2 (programación convexa) *Considérese el problema de programación convexa (CPP). Minimizar*

$$Z = f(\mathbf{x})$$

sujeto a

$$\mathbf{x} \in S$$

donde f es convexa y diferenciable y S es un conjunto convexo. Entonces $\mathbf{x}^* \in S$ es un mínimo global si y sólo si

$$\nabla f(\mathbf{x}^*)^T(\mathbf{x} - \mathbf{x}^*) \geq 0, \ \forall \mathbf{x} \in S$$

Demostración. Sea $\mathbf{x}$ una solución factible del PPC. Entonces

$$\mathbf{x}^* + \lambda(\mathbf{x} - \mathbf{x}^*) = \lambda\mathbf{x} + (1-\lambda)\mathbf{x}^* \in S, \forall\lambda \in [0,1]$$

por la convexidad del conjunto S. Esto implica que $\mathbf{x}^* + \lambda\mathbf{d} \in S, \forall\lambda \in (0,1)$, donde $\mathbf{d} = \mathbf{x} - \mathbf{x}^*$ es una dirección de descenso factible. Considérese la función de una variable definida en $\lambda \in [0,1]$ por

$$\varphi(\lambda) = f(\mathbf{x}^* + \lambda\mathbf{d}).$$

Por la hipótesis de que $\mathbf{x}^*$ es un óptimo global de f en S, φ tiene un mínimo en el punto $\lambda = 0$. Entonces, aplicando la regla de la cadena, se obtiene

$$0 \leq \varphi'(0) = \nabla f(\mathbf{x}^*)^T\mathbf{d}$$

con lo que queda probada esta implicación.

Recíprocamente, empleando el teorema 8.3, por la convexidad y diferenciabilidad de la función f en S se obtiene

$$f(\mathbf{x}) - f(\mathbf{x}^*) \geq \nabla f(\mathbf{x}^*)^T(\mathbf{x} - \mathbf{x}^*) \geq 0 \quad \forall \mathbf{x} \in S.$$

■

El teorema siguiente enuncia una condición suficiente para asegurar que todo punto de KKT sea un mínimo global.

Teorema 8.5 (suficiencia de las condiciones de Karush–Kuhn–Tucker). *Considérese el PPNL consistente en la minimización de*

$$Z = f(\mathbf{x})$$

sujeto a

$$\begin{array}{rcl} \mathbf{h}(\mathbf{x}) & = & \mathbf{0} \\ \mathbf{g}(\mathbf{x}) & \leq & \mathbf{0} \end{array}$$

Supóngase que existe una terna de vectores $(\bar{\mathbf{x}}, \bar{\boldsymbol{\mu}}, \bar{\boldsymbol{\lambda}})$ que cumple las CKKT. Sea $K^+ = \{k|\bar{\lambda}_k > 0\}$ y $K^- = \{k|\bar{\lambda}_k < 0\}$. Supóngase que $f(\mathbf{x})$, $g_i(\mathbf{x})$ para todo $i \in I(\bar{\mathbf{x}})$ y $h_k(\mathbf{x})$ para todo $k \in K^+$ son funciones convexas en $\mathbb{R}^n$, y $h_k(\mathbf{x})$ para todo $k \in K^-$ son funciones cóncavas en $\mathbb{R}^n$. Entonces $\bar{\mathbf{x}}$ es un mínimo global.

■

Demostración. Se mostrará que $\bar{\mathbf{x}}$ es una solución óptima, para ello se empleará el lema 8.2. Sea $\mathbf{x}$ una solución factible del PPNL. Entonces, de $g_i(\mathbf{x}) \leq 0$ y de $g_i(\bar{\mathbf{x}}) = 0$, para $i \in I(\bar{\mathbf{x}})$, se tiene $g_i(\mathbf{x}) \leq g_i(\bar{\mathbf{x}})$, y por la convexidad de g_i en S se sigue que

$$g_i(\bar{\mathbf{x}} + \lambda(\mathbf{x} - \bar{\mathbf{x}})) = g_i(\lambda\mathbf{x} + (1-\lambda)\bar{\mathbf{x}}) \leq \lambda g_i(\mathbf{x}) + (1-\lambda)g_i(\bar{\mathbf{x}}) \leq g_i(\bar{\mathbf{x}})$$

para todo $\lambda \in (0,1)$. Esto implica que la función g_i no crece cuando la variable $\bar{\mathbf{x}}$ se mueve en la dirección $\mathbf{d} = \mathbf{x} - \bar{\mathbf{x}}$. Entonces, se tiene que

$$\nabla g_i(\bar{\mathbf{x}})^T(\mathbf{x} - \bar{\mathbf{x}}) \leq \mathbf{0} \qquad (8.59)$$

Similarmente, de la convexidad de h_k para todo $k \in K^+$ y de la concavidad de las funciones h_k con $k \in K^-$, se tiene que

$$\nabla h_k(\bar{\mathbf{x}})^T(\mathbf{x} - \bar{\mathbf{x}}) \leq \mathbf{0}, \quad \forall k \in K^+ \qquad (8.60)$$
$$\nabla h_k(\bar{\mathbf{x}})^T(\mathbf{x} - \bar{\mathbf{x}}) \geq \mathbf{0}, \quad \forall k \in K^- \qquad (8.61)$$

Multiplicando (8.59), (8.60), y (8.61) por $\bar{\mu}_i \geq 0$, $\bar{\lambda}_k > 0$, y $\bar{\lambda}_k < 0$, respectivamente, y sumándolo, se obtiene

$$\left[\sum_{k \in K^+ \cup K^-} \bar{\lambda}_k \nabla h_k(\bar{\mathbf{x}}) + \sum_{i \in I(\bar{\mathbf{x}})} \mu_i \nabla g_i(\bar{\mathbf{x}})\right]^T (\mathbf{x} - \bar{\mathbf{x}}) \leq 0 \qquad (8.62)$$

Recuérdese que, por hipótesis, $(\bar{x}, \bar{\mu}, \bar{\lambda})$ es un punto de KKT. Multiplicando la condición (8.53) por $\mathbf{x} - \bar{\mathbf{x}}$ y teniendo en cuenta que $\bar{\mu}_i = 0$ para todo $i \notin I(\bar{\mathbf{x}})$, entonces (8.62) implica que

$$\nabla f(\bar{\mathbf{x}})^T(\mathbf{x} - \bar{\mathbf{x}}) \geq 0$$

y empleando el lema 8.2 se completa la demostración. ∎

Ejemplo 8.6 (casos especiales: problemas no restringidos). Para todos los problemas no restringidos las CKKT son condiciones necesarias de optimalidad. Sea la función $f : \mathbb{R}^n \to \mathbb{R}$ convexa en $\mathbb{R}^n$ y diferenciable en $\mathbf{x}^*$. Entonces $\mathbf{x}^*$ es un mínimo global del problema: Minimizar

$$Z = f(\mathbf{x}), \quad \mathbf{x} \in \mathbb{R}^n$$

si y sólo si $\nabla f(\mathbf{x}^*) = 0$. ∎

La clase más importante de PPNLs para la que las condiciones de KKT son siempre condiciones necesarias y suficientes de optimalidad son, ciertamente, los llamados problemas convexos (PC). Un PC consiste en encontrar el mínimo de una función convexa sobre un conjunto de restricciones de desigualdad que están definidas por funciones convexas, mientras que las restricciones de igualdad

están definidas por funciones afines. Más concretamente, se está interesado en el problema: Minimizar

$$Z = f(\mathbf{x})$$

sujeto a

$$\mathbf{g}(\mathbf{x}) \leq \mathbf{0}$$

donde f y $\mathbf{g}$ son convexas y continuamente diferenciables. Se denota por S la región factible definida por las restricciones.

Una representación más detallada de este problema se obtiene al considerar como restricción la función $\mathbf{h}(\mathbf{x}) = \mathbf{A}\mathbf{x} + \mathbf{b}$, donde $\mathbf{A}$ es una matriz $m \times n$, y $\mathbf{b}$ un vector $n \times 1$, que es simultáneamente cóncava y convexa; entonces $\mathbf{h}$ y $-\mathbf{h}$ son funciones convexas. La función $\mathbf{h}$ se denomina *función afín* (una función lineal más un término constante). Las restricciones de la forma $\mathbf{h}(\mathbf{x}) = \mathbf{0}$ donde $\mathbf{h}$ es una función afín son frecuentes en los problemas convexos. Lo anterior conduce a escribir los problemas convexos de la forma siguiente. Minimizar

$$Z = f(\mathbf{x})$$

sujeto a

$$\begin{array}{rcl} \mathbf{h}(\mathbf{x}) & = & \mathbf{0} \\ \mathbf{g}(\mathbf{x}) & \leq & \mathbf{0} \end{array}$$

donde las funciones f y $\mathbf{g}$ son convexas y continuamente diferenciables, y $\mathbf{h}$ es una función afín. Un caso especialmente importante es la programación lineal (estudiada en los capítulos anteriores) al que se puede aplicar todos los resultados de este capítulo.

El importante concepto de convexidad se ilustra mediante los dos ejemplos siguientes.

Ejemplo 8.7 (ajuste mínimo cuadrático). En muchos campos de la ciencia y de la tecnología se tienen resultados experimentales que relacionan cierto par de funciones $r(x)$ y $s(y)$ definidas respectivamente sobre las variables x e y. La relación entre estas cantidades es desconocida, pero se posee información sobre un conjunto de pares de datos $\{(r(x_i), s(y_i)) | i = 1, 2, \ldots, n\}$. En muchas ocasiones se desea encontrar la curva que minimice el error cuadrático medio de la coordenada $s(y)$. Si se ajusta un modelo lineal al conjunto de datos, se tendría que encontrar la pendiente α y el término constante β que minimizan la función objetivo

$$f(\alpha, \beta) = \sum_i \left(\alpha r(x_i) + \beta - s(y_i)\right)^2$$

Ambos parámetros α y β no están restringidos ni en signo, ni de ningún otro modo, lo que conduce a un PPNL no restringido. Del hecho de que la función objetivo f tienda a infinito cuando $(\alpha, \beta) \to \infty$ y, por otro lado, $f \geq 0$, garantiza que el PPNL admite soluciones óptimas. La función f es convexa en las variables

(α, β), incluso más, es estrictamente convexa. Consecuentemente, el problema de optimización tiene una única solución que puede ser hallada empleando las condiciones de KKT. En el caso de problemas no restringidos, estas condiciones imponen que las derivadas parciales de la función f respecto a las variables α y β deban ser cero:

$$\begin{array}{rcl} \dfrac{\partial f(\alpha,\beta)}{\partial \alpha} & = & \sum_i (\alpha r(x_i) + \beta - s(y_i)) r(x_i) = 0 \\ \dfrac{\partial f(\alpha,\beta)}{\partial \alpha} & = & \sum_i (\alpha r(x_i) + \beta - s(y_i)) = 0 \end{array}$$

o después de agrupar términos, se llega al sistema de ecuaciones lineales

$$\begin{array}{rcl} \displaystyle\sum_i r(x_i)^2 \alpha + \sum_i r(x_i) \beta & = & \displaystyle\sum_i r(x_i) s(y_i) \\ \displaystyle\sum_i r(x_i) \alpha + n\beta & = & \displaystyle\sum_i s(y_i) \end{array}$$

La solución de este sistema es

$$\begin{array}{rcl} \alpha & = & \dfrac{n \sum_i r(x_i) s(y_i) - \sum_i r(x_i) \sum_i s(y_i)}{(n \sum_i r(x_i)^2 - (\sum_i r(x_i))^2)} \\ \beta & = & \dfrac{\sum_i r(x_i)^2 \sum_i s(y_i) - \sum_i r(x_i) \sum_i r(x_i) s(y_i)}{(n \sum_i r(x_i)^2 - (\sum_i r(x_i))^2)} \end{array}$$

■

Ejemplo 8.8 (problema de Cobb–Douglas). El nivel de producción de un determinado producto depende del nivel de inversión en materiales, que se denota por x_1, y de los recursos humanos empleados, denotado por x_2. La función de Cobb–Douglas determina el nivel de producción en función de las variables x_1 y x_2, y ésta tiene la forma:

$$f(x_1, x_2) = x_1^{\alpha} x_2^{\beta}$$

donde $\alpha > 0$, $\beta > 0$, y $\alpha + \beta \leq 1$. Se supone que existe una restricción presupuestaria que limita el nivel de inversión. Sea C el presupuesto total disponible, y sea a el coste monetario de cada unidad de recursos humanos, esta restricción se escribe

$$x_1 + a x_2 = C. \qquad (8.63)$$

Un empresario desea determinar la distribución del presupuesto entre bienes materiales y recursos humanos de modo que se maximice el nivel de producción, lo que conduce al problema de optimización: Maximizar

$$Z = x_1^{\alpha} x_2^{\beta}$$

sujeto a

$$\begin{array}{rrcl} x_1 & +ax_2 & = & C \\ -x_1 & & \leq & 0 \\ & -x_2 & \leq & 0 \end{array}$$

Puesto que

$$\max_{\mathbf{x}\in S} f(\mathbf{x}) = -\min_{\mathbf{x}\in S}(-f(\mathbf{x}))$$

el problema anterior puede ser formulado como sigue. Minimizar

$$Z = -x_1^{\alpha} x_2^{\beta}$$

sujeto a

$$\begin{array}{rrcl} x_1 & +ax_2 & = & C \\ -x_1 & & \leq & 0 \\ & -x_2 & \leq & 0 \end{array}$$

Este es un PC debido a que la restricción es lineal y a que la función objetivo es convexa, cuestión que se mostrará empleando el teorema 8.4. El vector gradiente es

$$\nabla f(x_1, x_2) = \begin{pmatrix} -\alpha x_1^{\alpha-1} x_2^{\beta} \\ -\beta x_1^{\alpha} x_2^{\beta-1} \end{pmatrix}$$

y la matriz hessiana es

$$\nabla^2 f(x_1, x_2) = \begin{pmatrix} -\alpha(\alpha-1) x_1^{\alpha-2} x_2^{\beta} & -\alpha\beta x_1^{\alpha-1} x_2^{\beta-1} \\ -\alpha\beta x_1^{\alpha-1} x_2^{\beta-1} & -\beta(\beta-1) x_1^{\alpha} x_2^{\beta-2} \end{pmatrix}$$

Empleando el criterio del determinante, según se vio en la definición 8.9, se probará que la matriz hessiana es definida positiva. Por hipótesis, el parámetro α cumple $0 < \alpha < 1$ y $-\alpha(\alpha-1) > 0$; en la región factible, $x_1 \geq 0$ y $x_2 \geq 0$, entonces $x_1^{\alpha-2} x_2^{\beta} \geq 0$. Por tanto, $\det(\nabla^2 f(x_1, x_2)) = \alpha\beta(1-\alpha-\beta) x_1^{2(\alpha-1)} x_2^{2(\beta-1)}$, y es no negativo en la región factible. En consecuencia, $\nabla^2 f(x_1, x_2)$ es definida positiva y el teorema 8.5 justifica que cualquier punto de KKT es una solución óptima del problema. La condición de optimalidad mediante el Lagrangiano se expresa

$$\begin{pmatrix} -\alpha x_1^{\alpha-1} x_2^{\beta} \\ -\beta x_1^{\alpha} x_2^{\beta-1} \end{pmatrix} + \mu_1 \begin{pmatrix} 1 \\ 0 \end{pmatrix} + \mu_2 \begin{pmatrix} 0 \\ 1 \end{pmatrix} + \lambda \begin{pmatrix} 1 \\ a \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

Se supondrá que $x_1 > 0$ y $x_2 > 0$, y empleando la condición de complementariedad se llega a $\mu_1 = \mu_2 = 0$. Entonces, el sistema de ecuaciones

$$\begin{array}{rcl} -\alpha x_1^{\alpha-1} x_2^{\beta} + \lambda & = & 0 \\ -\beta x_1^{\alpha} x_2^{\beta-1} + \lambda a & = & 0 \end{array}$$

conduce a

$$-\alpha x_1^{\alpha-1}x_2^{\beta} = -\frac{\beta}{a}x_1^{\alpha}x_2^{\beta-1}$$

Dividiendo por $x_1^{\alpha-1}x_2^{\beta-1} > 0$, resulta

$$x_2 = \frac{\beta}{a\alpha}x_1$$

y empleando (8.63), se obtiene finalmente

$$\begin{array}{rcl} x_1 & = & \dfrac{\beta}{\alpha+\beta}C \\ x_2 & = & \dfrac{\alpha}{a(\alpha+\beta)}C \end{array}$$

■

8.4 Teoría de la dualidad

La dualidad juega un papel crucial en el desarrollo de algoritmos para la resolución de problemas de programación no lineal (véase, por ejemplo, Bazaraa et al. [9] y Luenberger [68]). En esta sección se generalizan los resultados que se obtuvieron para la programación lineal a los problemas de programación convexa, y en el próximo capítulo estos resultados se emplean para desarrollar métodos que obtienen soluciones aproximadas de los PPNL.

Considérese el problema no lineal siguiente como problema *primal* (P): Minimizar

$$Z_P = f(\mathbf{x}) \tag{8.64}$$

sujeto a

$$\begin{array}{rcl} \mathbf{h}(\mathbf{x}) & = & \mathbf{0} \\ \mathbf{g}(\mathbf{x}) & \le & \mathbf{0} \end{array} \tag{8.65}$$

donde $f : \mathbb{R}^n \to \mathbb{R}$, $\mathbf{h} : \mathbb{R}^n \to \mathbb{R}^\ell$, $\mathbf{g} : \mathbb{R}^n \to \mathbb{R}^m$.

El *problema dual* requiere la introducción de la llamada *función dual* definida por

$$\theta(\boldsymbol{\lambda}, \boldsymbol{\mu}) \quad = \quad \text{Inf}_{\mathbf{x}} \ \{f(\mathbf{x}) + \boldsymbol{\lambda}^T\mathbf{h}(\mathbf{x}) + \boldsymbol{\mu}^T\mathbf{g}(\mathbf{x})\} \tag{8.66}$$

El *problema dual* (D) está definido como sigue. Maximizar

$$Z_D = \theta(\boldsymbol{\lambda}, \boldsymbol{\mu})$$

sujeto a

$$\boldsymbol{\mu} \ge \mathbf{0}$$

Empleando la función lagrangiana

$$\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = f(\mathbf{x}) + \boldsymbol{\lambda}^T\mathbf{h}(\mathbf{x}) + \boldsymbol{\mu}^T\mathbf{g}(\mathbf{x})$$

se puede reescribir D como

$$\max_{\boldsymbol{\lambda},\boldsymbol{\mu};\boldsymbol{\mu}\geq 0}\{\text{ ínfimo}_{\mathbf{x}}\ \ \mathcal{L}(\mathbf{x},\boldsymbol{\lambda},\boldsymbol{\mu})\} \tag{8.67}$$

Siempre se supondrá que las funciones $f, \mathbf{h}$, y $\mathbf{g}$ son de tal forma que el ínfimo de la función lagrangiana siempre se alcanza en algún punto $\mathbf{x}$, por lo que el operador "ínfimo" en (8.66) y (8.67) puede ser reemplazado por el operador "mínimo". Entonces, el problema (8.67) se denomina *problema dual max–min.*

En esta situación, si se denota por $\mathbf{x}(\boldsymbol{\lambda},\boldsymbol{\mu})$ un punto en el que el se alcance el valor mínimo de la función lagrangiana (considerada como una función de los multiplicadores), se podrá escribir

$$\theta(\boldsymbol{\lambda},\boldsymbol{\mu}) = f(\mathbf{x}(\boldsymbol{\lambda},\boldsymbol{\mu})) + \boldsymbol{\lambda}^T\mathbf{h}(\mathbf{x}(\boldsymbol{\lambda},\boldsymbol{\mu})) + \boldsymbol{\mu}^T\mathbf{g}(\mathbf{x}(\boldsymbol{\lambda},\boldsymbol{\mu}))$$

Por otro lado, bajo la hipótesis de convexidad de las funciones f y $\mathbf{g}$, de que $\mathbf{h}$ es una función afín y que $\boldsymbol{\mu}$ es no negativo, el lagrangiano es convexo en $\mathbf{x}$, y por tanto, el gradiente se debe anular en los mínimos globales

$$\nabla f(\mathbf{x}(\boldsymbol{\lambda},\boldsymbol{\mu})) + \boldsymbol{\lambda}^T\nabla\mathbf{h}(\mathbf{x}(\boldsymbol{\lambda},\boldsymbol{\mu})) + \boldsymbol{\mu}^T\nabla\mathbf{g}(\mathbf{x}(\boldsymbol{\lambda},\boldsymbol{\mu})) = \mathbf{0} \tag{8.68}$$

Estas dos últimas identidades se pueden emplear para derivar expresiones para el gradiente y el hessiano de la función dual. Aplicando la regla de la cadena resulta

$$\begin{aligned}\nabla_{\boldsymbol{\mu}}\theta(\boldsymbol{\lambda},\boldsymbol{\mu}) = {} & [\nabla f(\mathbf{x}(\boldsymbol{\lambda},\boldsymbol{\mu})) + \boldsymbol{\lambda}^T\nabla\mathbf{h}(\mathbf{x}(\boldsymbol{\lambda},\boldsymbol{\mu})) + \boldsymbol{\mu}^T\nabla\mathbf{g}(\mathbf{x}(\boldsymbol{\lambda},\boldsymbol{\mu}))]^T\nabla_{\boldsymbol{\mu}}\mathbf{x}(\boldsymbol{\lambda},\boldsymbol{\mu}) \\ & +\mathbf{g}(\mathbf{x}(\boldsymbol{\lambda},\boldsymbol{\mu}))\end{aligned}$$

y por (8.68)

$$\nabla_{\boldsymbol{\mu}}\theta(\boldsymbol{\lambda},\boldsymbol{\mu}) = \mathbf{g}(\mathbf{x}(\boldsymbol{\lambda},\boldsymbol{\mu})) \tag{8.69}$$

Similarmente

$$\nabla_{\boldsymbol{\lambda}}\theta(\boldsymbol{\lambda},\boldsymbol{\mu}) = \mathbf{h}(\mathbf{x}(\boldsymbol{\lambda},\boldsymbol{\mu})) \tag{8.70}$$

esto es, el gradiente de la función dual es el vector de las restricciones evaluado en la solución óptima del lagrangiano.

Mediante la diferenciación de (8.69) y (8.70) respecto a $\boldsymbol{\mu}$ y $\boldsymbol{\lambda}$ respectivamente, se obtiene el hessiano de la función dual

$$\nabla^2_{\boldsymbol{\mu}}\theta(\boldsymbol{\lambda},\boldsymbol{\mu}) = \nabla_{\mathbf{x}}\mathbf{g}(\mathbf{x}(\boldsymbol{\lambda},\boldsymbol{\mu}))\ \ \nabla_{\boldsymbol{\mu}}\mathbf{x}(\boldsymbol{\lambda},\boldsymbol{\mu}) \tag{8.71}$$

y

$$\nabla^2_{\boldsymbol{\lambda}}\theta(\boldsymbol{\lambda},\boldsymbol{\mu}) = \nabla_{\mathbf{x}}\mathbf{h}(\mathbf{x}(\boldsymbol{\lambda},\boldsymbol{\mu}))\ \ \nabla_{\boldsymbol{\lambda}}\mathbf{x}(\boldsymbol{\lambda},\boldsymbol{\mu}) \tag{8.72}$$

Para obtener una expresión para $\nabla_{\boldsymbol{\mu}}\mathbf{x}(\boldsymbol{\lambda},\boldsymbol{\mu})$ y $\nabla_{\boldsymbol{\lambda}}\mathbf{x}(\boldsymbol{\lambda},\boldsymbol{\mu})$, se deriva (8.68) con respecto a $\boldsymbol{\mu}$ y $\boldsymbol{\lambda}$, respectivamente. El resultado es

$$\nabla^2_{\mathbf{x}}\mathcal{L}(\mathbf{x}(\boldsymbol{\lambda},\boldsymbol{\mu}),\boldsymbol{\lambda},\boldsymbol{\mu})\ \nabla_{\boldsymbol{\mu}}\mathbf{x}(\boldsymbol{\lambda},\boldsymbol{\mu}) + \nabla_{\mathbf{x}}\mathbf{g}(\mathbf{x}(\boldsymbol{\lambda},\boldsymbol{\mu})) = \mathbf{0}$$

y se obtiene una ecuación similar cambiando $\boldsymbol{\lambda}$ por $\boldsymbol{\mu}$ y $\mathbf{h}$ por $\mathbf{g}$. Teniendo en cuenta estas fórmulas en (8.71) y (8.72), se llega a

$$\nabla^2_{\boldsymbol{\mu}}\theta(\boldsymbol{\lambda},\boldsymbol{\mu}) = -\nabla_{\mathbf{x}}\mathbf{g}(\mathbf{x}(\boldsymbol{\lambda},\boldsymbol{\mu}))[\nabla^2_{\mathbf{x}}\mathcal{L}(\mathbf{x}(\boldsymbol{\lambda},\boldsymbol{\mu}),\boldsymbol{\lambda},\boldsymbol{\mu})]^{-1}\ \nabla_{\mathbf{x}}\mathbf{g}(\mathbf{x}(\boldsymbol{\lambda},\boldsymbol{\mu}))$$

y a la fórmula paralela para $\nabla^2_{\boldsymbol{\lambda}}\theta(\boldsymbol{\lambda},\boldsymbol{\mu})$. Finalmente, para las derivadas mixtas segundas

$$\nabla^2_{\boldsymbol{\lambda}\,\boldsymbol{\mu}}\theta(\boldsymbol{\lambda},\boldsymbol{\mu})$$

los mismos cálculos conducen a la expresión

$$\nabla^2_{\boldsymbol{\lambda}\,\boldsymbol{\mu}}\theta(\boldsymbol{\lambda},\boldsymbol{\mu}) = -\nabla_{\mathbf{x}}\mathbf{g}(\mathbf{x}(\boldsymbol{\lambda},\boldsymbol{\mu}))[\nabla^2_{\mathbf{x}}\mathcal{L}(\mathbf{x}(\boldsymbol{\lambda},\boldsymbol{\mu}),\boldsymbol{\lambda},\boldsymbol{\mu})]^{-1}\ \nabla_{\mathbf{x}}\mathbf{h}(\mathbf{x}(\boldsymbol{\lambda},\boldsymbol{\mu}))$$

y similarmente

$$\nabla^2_{\boldsymbol{\lambda}\,\boldsymbol{\mu}}\theta(\boldsymbol{\lambda},\boldsymbol{\mu}) = -\nabla_{\mathbf{x}}\mathbf{h}(\mathbf{x}(\boldsymbol{\lambda},\boldsymbol{\mu}))[\nabla^2_{\mathbf{x}}\mathcal{L}(\mathbf{x}(\boldsymbol{\lambda},\boldsymbol{\mu}),\boldsymbol{\lambda},\boldsymbol{\mu})]^{-1}\ \nabla_{\mathbf{x}}\mathbf{g}(\mathbf{x}(\boldsymbol{\lambda},\boldsymbol{\mu}))$$

Todas estas fórmulas son relevantes para los métodos computacionales, como se verá en el capítulo siguiente.

El siguiente teorema muestra que cualquier valor del problema dual es una cota inferior del valor óptimo del problem primal. Este resultado puede ser aplicado como criterio de terminación de un algoritmo iterativo para el que estos valores estén disponibles.

Teorema 8.6 (dualidad débil). *Cualquier solución factible* $\mathbf{x}$ *del problema primal* (*8.65*) *y cualquier solución del problema dual* (*8.66*), *cumplen*

$$f(\mathbf{x}) \geq \theta(\boldsymbol{\lambda},\boldsymbol{\mu}) \tag{8.73}$$

■

Demostración. De la definición de θ, para cualquier solución $\mathbf{x}$ y para cualquier solución $\boldsymbol{\mu} \geq 0$ y $\boldsymbol{\lambda} \in \mathbb{R}^{\ell}$, se tiene

$$\theta(\boldsymbol{\lambda},\boldsymbol{\mu}) = \ \text{Inf}_{\mathbf{y}}\left[f(\mathbf{y}) + \boldsymbol{\lambda}^T\mathbf{h}(\mathbf{y}) + \boldsymbol{\mu}^T\mathbf{g}(\mathbf{y})\right] \leq f(\mathbf{x}) + \boldsymbol{\lambda}\mathbf{h}(\mathbf{x}) + \boldsymbol{\mu}^T\mathbf{g}(\mathbf{x}) \leq f(\mathbf{x})$$

■

Si la región factible de los problemas primal y dual es no vacía, entonces (8.73) implica que $\sup\theta$ e $\inf f$, sobre sus respectivos conjuntos factibles, son finitos. Desde luego, ni el *supremo* ni el *ínfimo* se tienen que alcanzar necesariamente en sus respectivos conjuntos factibles. Si así lo fuese, la relación (8.73) garantizaría que serían soluciones óptimas del problema dual y primal respectivamente.

Un corolario de gran interés práctico es el siguiente.

Corolario 8.2 (optimalidad dual y primal)
1. Se cumple lo siguiente

$$\sup\{\theta(\boldsymbol{\lambda},\boldsymbol{\mu})|\boldsymbol{\mu} \geq \mathbf{0}\} \leq \inf\{f(\mathbf{x})|\mathbf{h}(\mathbf{x}) = \mathbf{0}, \mathbf{g}(\mathbf{x}) \leq \mathbf{0}\} \tag{8.74}$$

2. Si $f(\mathbf{x}^*) = \theta(\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*)$ *para alguna solución factible* $\mathbf{x}^*$ *del problema primal (8.65) y para alguna solución factible* $(\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*)$ *del problema dual (8.67), entonces* $\mathbf{x}^*$ *y* $(\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*)$ *son, respectivamente, soluciones del problema primal y dual.*

3. Si $\sup\{\theta(\boldsymbol{\lambda}, \boldsymbol{\mu}) : \boldsymbol{\mu} \geq \mathbf{0}\} = +\infty$*, entonces el problema primal no tiene soluciones factibles.*

4. Si $\inf\{f(\mathbf{x}) : \mathbf{h}(\mathbf{x}) = \mathbf{0}, \mathbf{g}(\mathbf{x}) \leq \mathbf{0}\} = -\infty$*, entonces* $\theta(\boldsymbol{\lambda}, \boldsymbol{\mu}) = -\infty$ *para cada* $\boldsymbol{\lambda} \in \mathbb{R}^\ell$ *y* $\boldsymbol{\mu} \geq \mathbf{0}$*.*

Si un vector de multiplicadores resuelve el problema dual y no existe la llamada *holgura dual*, esto es, si la relación (8.74) se satisface con igualdad, entonces, las soluciones del problema lagrangiano asociado con los multiplicadores son soluciones del problema primal. Este resultado permite resolver el problema primal mediante la resolución del problema dual. La cuestión fundamental es encontrar condiciones que garanticen la no existencia de la holgura dual, como ocurre en los programas convexos.

Teorema 8.7 (Teorema de dualidad para problemas convexos). *Considérese el problema convexo (8.64)–(8.65). Si* $\mathbf{x}^*$ *resuelve el problema primal, su vector asociado de multiplicadores* $(\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*)$ *resuelve el problema dual, y* $\boldsymbol{\mu}^*\mathbf{g}(\mathbf{x}^*) = 0$*. Recíprocamente, si* $(\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*)$ *resuelve el problema dual, y existe una solución,* $\mathbf{x}^*$*, del problema lagrangiano asociado a este vector de multiplicadores que cumple* $\boldsymbol{\mu}^*\mathbf{g}(\mathbf{x}^*) = 0$*, entonces* $\mathbf{x}^*$ *es una solución del problema primal.* ∎

Demostración. Si $\mathbf{x}^*$ resuelve el problema primal y $(\boldsymbol{\mu}^*, \boldsymbol{\lambda}^*)$ es su vector asociado de multiplicadores, se sabe que la solución es un punto de KKT

$$\begin{array}{rcl} \nabla f(\mathbf{x}^*) + \boldsymbol{\lambda}^*\nabla \mathbf{h}(\mathbf{x}^*) + \boldsymbol{\mu}^*\nabla \mathbf{g}(\mathbf{x}^*) & = & \mathbf{0} \\ \boldsymbol{\mu}^*\mathbf{g}(\mathbf{x}^*) & = & \mathbf{0} \\ \boldsymbol{\mu}^* & \geq & \mathbf{0} \\ \mathbf{g}(\mathbf{x}^*) & \leq & \mathbf{0} \\ \mathbf{h}(\mathbf{x}^*) & = & \mathbf{0} \end{array}$$

Seguidamente, se considera el problema de minimización definido por $\theta(\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*)$

$$\theta(\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*) = \ \text{Inf}_{\mathbf{x}} \ \{f(\mathbf{x}) + \boldsymbol{\lambda}^{*T}\mathbf{h}(\mathbf{x}) + \boldsymbol{\mu}^{*T}\mathbf{g}(\mathbf{x})\}$$

La condición derivada del anterior sistema de KKT es

$$\nabla f(\mathbf{x}^*) + \boldsymbol{\lambda}^*\nabla \mathbf{h}(\mathbf{x}^*) + \boldsymbol{\mu}^*\nabla \mathbf{g}(\mathbf{x}^*) = \mathbf{0}$$

junto a la condición de convexidad de la función objetivo (obsérvese que $\boldsymbol{\mu}^* \geq \mathbf{0}$)

$$f(\mathbf{x}) + \boldsymbol{\lambda}^{*T}\mathbf{h}(\mathbf{x}) + \boldsymbol{\mu}^{*T}\mathbf{g}(\mathbf{x})$$

implica que $\mathbf{x}^*$ es un mínimo global del problema determinado por $\theta(\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*)$ y por tanto

$$\theta(\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*) = f(\mathbf{x}^*)$$

Además, por el corolario 8.2 el par $(\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*)$ es una solución óptima del problema dual. Nótese que la condición

$$\boldsymbol{\mu}^*\mathbf{g}(\mathbf{x}^*) = \mathbf{0}$$

también se cumple.

Recíprocamente, supóngase que el par $(\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*)$ es óptimo para el problema dual, y que $\mathbf{x}^*$ es su correspondiente solución para el problema lagrangiano asociado, tal que $(\boldsymbol{\mu}^*)^T\mathbf{g}(\mathbf{x}^*) = 0$. Aplicando las condiciones de KKT al problema dual, se obtiene sin dificultad que

$$\nabla_{\boldsymbol{\mu}}\theta(\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*) \leq \mathbf{0}, \quad \nabla_{\boldsymbol{\lambda}}\theta(\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*) = \mathbf{0}$$

Pero, por (8.69) y (8.70), se obtiene

$$\mathbf{g}(\mathbf{x}^*) \leq \mathbf{0}; \;\; \mathbf{h}(\mathbf{x}^*) = \mathbf{0}$$

Esto implica que $\mathbf{x}^*$ es factible para el primal. Entonces

$$\theta(\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*) = f(\mathbf{x}^*) + \boldsymbol{\lambda}^{*T}\mathbf{h}(\mathbf{x}^*) + \boldsymbol{\mu}^{*T}\mathbf{g}(\mathbf{x}^*) = f(\mathbf{x}^*)$$

y por el corolario 8.2 resulta probado el teorema. ∎

El ejemplo siguiente ilustra el concepto de dualidad.

Ejemplo 8.9 (problema dual). Considérese el problema primal (P): Minimizar

$$Z_P = x_1^2 + x_2^2$$

sujeto a

$$\begin{aligned} x_1 + x_2 &\leq 4 \\ -x_1 &\leq 0 \\ -x_2 &\leq 0 \end{aligned}$$

Este es un problema de programación convexa con una única solución óptima que es $(x_1^*, x_2^*) = (0, 0)$ y el valor mínimo de la función objetivo es 0. Ahora se formula su problema dual (D). Se necesita primeramente encontrar la función dual:

$$\begin{aligned} \theta(\mu_1, \mu_2, \mu_3) &= \min_{\mathbf{x} \in \mathbf{R}^2} \{x_1^2 + x_2^2 + \mu_1(x_1 + x_2 - 4) + \mu_2(-x_1) + \mu_3(-x_2)\} \\ &= \min_{x_1}\{x_1^2 + (\mu_1 - \mu_2)x_1\} + \min_{x_2}\{x_2^2 + (\mu_1 - \mu_3)x_2\} - 4\mu_1 \end{aligned}$$

Realizando diversas operaciones elementales se obtiene explícitamente la función lagrangiana

$$\theta(\boldsymbol{\mu}) = -\frac{1}{4}[(\mu_1 - \mu_2)^2 + (\mu_1 - \mu_3)^2] - 4\mu_1$$

y el problema dual (D) es

$$\sup\{\theta(\boldsymbol{\mu})|\boldsymbol{\mu} \geq 0\}$$

Nótese que la función dual es cóncava. El problema dual alcanza su máximo en $\mu_1^* = \mu_2^* = \mu_3^* = 0$ y $Z_D^* = \theta(\boldsymbol{\mu}^*) = 0$. En este ejemplo no existe holgura dual, como ya se sabía a priori, por tratarse de un problema convexo. ∎

Es importante destacar en este punto que se puede considerar una dualidad parcial, en el sentido de que sólo algunas de las restricciones son dualizadas; las que no lo son, se incorporan en la definición de la función dual. Este concepto es la base de la descomposición lagrangiana, que será discutida en el capítulo 9.

8.5 Ilustración práctica de la dualidad y separabilidad

Para ilustrar la relación entre el problema primal y el dual, considérese que un producto es demandado y consumido bajo las siguientes condiciones:

1. Diferentes productores compiten en un mercado.
2. La demanda en cada periodo en el que se ha dividido el horizonte temporal de la planificación debe ser satisfecha.
3. No existe capacidad de almacenamiento del producto.
4. Los productores tienen diferentes costes y restricciones de producción.

Se considerarán dos enfoques diferentes del problema: (i) el esquema centralizado o método *primal* y el del (ii) mercado competitivo o esquema *dual*.

8.5.1 Esquema centralizado o método primal

Un esquema centralizado (primal) para este problema es el siguiente. Los productores eligen un operador del sistema que se encarga de satisfacer la demanda en cada periodo al mínimo coste posible.

Para este fin, el operador del sistema:

1. Sabe los costes de producción de cada uno de los productores y su capacidad de producción
2. Conoce la demanda del bien en cada periodo
3. Tiene la capacidad de regular la producción de cada productor

Los principales elementos de este problema son

1. **Datos.**

$c_{ti}(p_{ti})$: costes de producción del productor i en el periodo t para producir p_{ti}

Π_i: región factible del productor i

d_t: demanda para el periodo t

T: número de periodos considerados en la planificación

I: número de productores

2. **Variables.**

   p_{ti}: cantidad del bien producida por el productor i en el periodo t

3. **Restricciones.**

$$\sum_{i=1}^{I} p_{ti} = d_t, \qquad t = 1, \ldots, T \tag{8.75}$$

$$p_{ti} \in \Pi_i, \qquad t = 1, \ldots, T, i = 1, \ldots, I \tag{8.76}$$

   Estas restricciones fuerzan a que en cada periodo la demanda se satisfaga totalmente, cumpliéndose las restricciones operacionales de cada productor.

4. **Función objetivo.** El operador tiene el objetivo de minimizar los costes totales de producción, lo que conduce a minimizar la función siguiente

$$\sum_{t=1}^{T} \sum_{i=1}^{I} c_{ti}(p_{ti}), \tag{8.77}$$

   que es la suma de los costes de producción de todos los productores en todos los periodos considerados.

Entonces, el problema primal es el siguiente. Minimizar

$$\sum_{t=1}^{T} \sum_{i=1}^{I} c_{ti}(p_{ti}) \tag{8.78}$$

sujeto a

$$\sum_{i=1}^{I} p_{ti} = d_t, \qquad t = 1, \ldots, T \tag{8.79}$$

$$p_{ti} \in \Pi_i, \qquad t = 1, \ldots, T, i = 1, \ldots, I \tag{8.80}$$

La solución de este problema proporciona los niveles óptimos de producción para cada productor. Entonces, el operador del sistema es el encargado de comunicar a cada productor su producción para cada periodo.

Nótese que este esquema es centralizado debido a que el operador del sistema tiene conocimiento de los costes de producción y la capacidad de producción de todos los productores del sistema. Además, el operador del sistema tiene la capacidad de imponer la producción (óptima) a los productores.

Esquema dual

Se analizará en esta sección la estructura del problema dual del problema anterior de minimización de costes de producción (problema primal) (8.77)–(8.80).

Si se dualizan (parcialmente) las restricciones (8.79); la función lagrangiana tiene, por tanto, la forma

$$\mathcal{L}(p_{ti}, \lambda_t) = \sum_{t=1}^{T} \sum_{i=1}^{I} c_{ti}(p_{ti}) + \sum_{t=1}^{T} \lambda_t \left(d_t - \sum_{i=1}^{I} p_{ti}. \right) \tag{8.81}$$

Organizando términos, la función lagrangiana puede ser expresada como

$$\mathcal{L}(p_{ti}, \lambda_t) = \sum_{i=1}^{I} \left[\sum_{t=1}^{T} (c_{ti}(p_{ti}) - \lambda_t \; p_{ti}) \right] + \sum_{t=1}^{T} \lambda_t \; d_t \tag{8.82}$$

La evaluación de la función dual para un valor dado de los multiplicadores de Lagrange $\lambda_t = \widehat{\lambda}_t$, $t = 1, \dots, T$, se obtiene resolviendo el problema: Maximizar

$$\mathcal{L}(p_{ti}, \widehat{\lambda}_t) \tag{8.83}$$

sujeto a

$$p_{ti} \quad \in \quad \Pi_i, \qquad t = 1, \dots, T, i = 1, \dots, I \tag{8.84}$$

Empleando la expresión explícita (8.82) para la función lagrangiana, el problema anterior se formula como la maximización de

$$\sum_{i=1}^{I} \left[\sum_{t=1}^{T} \left(c_{ti}(p_{ti}) - \widehat{\lambda}_t \; p_{ti} \right) \right] + \sum_{t=1}^{T} \widehat{\lambda}_t \; d_t \tag{8.85}$$

sujeto a

$$p_{ti} \quad \in \quad \Pi_i, \qquad t = 1, \dots, T, i = 1, \dots, I \tag{8.86}$$

Teniendo en cuenta que el término $\sum_{t=1}^{T} \widehat{\lambda}_t \; d_t$ es constante, el problema anterior se reduce a maximizar

$$\sum_{i=1}^{I} \left[\sum_{t=1}^{T} \left(c_{ti}(p_{ti}) - \widehat{\lambda}_t \; p_{ti} \right) \right] \tag{8.87}$$

sujeto a

$$p_{ti} \quad \in \quad \Pi_i, \qquad t = 1, \dots, T, i = 1, \dots, I \tag{8.88}$$

Este problema tiene la importante propiedad de la separabilidad; lo que permite descomponerlo en una colección de problemas (uno por cada productor) de menor dimensión. El problema individual del productor j consiste en maximizar

$$\sum_{t=1}^{T} \left(\widehat{\lambda}_t p_{tj} - c_{tj}(p_{tj}) \right) \tag{8.89}$$

sujeto a

$$p_{tj} \quad \in \quad \Pi_j, \qquad t = 1, \ldots, T \tag{8.90}$$

Entonces, la evaluación de la función dual se consigue resolviendo los problemas (8.89)-(8.90) para el conjunto de los productores I.

¿Qué es lo relevante en la interpretación del problema (8.89)–(8.90)? Representa la maximización del beneficio del productor j sujeto a restricciones de producción. Los multiplicadores de Lagrange λ_t, $t = 1, \ldots, T$ son interpretados como el precio de venta del producto en cada periodo $t = 1, \ldots, T$.

Se puede concluir que para evaluar la función dual cada productor debe planificar su propia producción de modo que maximice su beneficio, y entonces el operador, para calcular su valor, suma los beneficios de los productores más la constante $\sum_{t=1}^{T} \widehat{\lambda}_t \; d_t$.

Teorema de dualidad y solución del problema dual

El teorema de dualidad establece que, bajo las hipótesis previamente establecidas de convexidad, no existe holgura dual. Además, los multiplicadores de Lagrange óptimos λ_t^*, $t = 1, \ldots, T$, conducen a los valores de producción óptimos p_{ti}^*, $t = 1, \ldots, T$, $i = 1, \ldots, I$. Como consecuencia, se puede resolver el problema primal mediante la resolución del dual.

La teoría de dualidad establece que la función dual es cóncava bajo ciertas condiciones muy generales, lo que permite resolver este problema empleando un método sencillo, como es el método de mayor ascenso (véase el capítulo 9). Este procedimiento requiere conocer el gradiente de la función dual. La teoría de la dualidad da expresiones sencillas para el gradiente de la función dual. Estas expresiones han sido obtenidas anteriormente. Para la producción p_{ti}, $t = 1, \ldots, T$, $i = 1, \ldots, I$, las componentes del vector gradiente de la función dual son

$$d_t - \sum_{i=1}^{I} p_{ti} \qquad t = 1, \ldots, T \tag{8.91}$$

Esto es, la componente t del gradiente de la función dual es el desajuste entre la oferta y la demanda en la solución actual para el periodo t.

El método de mayor ascenso aplicado al problema dual consta de los siguientes pasos:

1. El operador establece unos valores iniciales para los multiplicadores de Lagrange (precios de venta del producto en cada periodo).

2. Empleando los multiplicadores anteriores, cada productor establece su propia producción, de modo que maximice su beneficio.

3. El operador calcula el desajuste entre la demanda y la oferta del bien en cada periodo, lo que constituye las componentes del vector gradiente.

4. Si la norma del gradiente de la función dual no es suficientemente pequeña, el operador suma a los multiplicadores actuales un múltiplo (positivo) del vector gradiente y repite el procedimiento comenzando en el paso 2. En caso contrario se para, la solución actual está próxima a la solución del problema dual.

8.5.2 Mercado competitivo o esquema dual

El algoritmo anterior puede ser interpretado en un contexto de un mercado competitivo del siguiente modo

1. El operador del mercado propone a los productores el precio inicial ($k = 1$) de venta en cada periodo, $\lambda_t^{(k)}$, $t = 1, \dots, T$.

2. Cada productor j busca su máximo beneficio mediante la resolución del problema: Maximizar

$$\sum_{t=1}^{T} \left(\lambda_t^{(k)} p_{tj} - c_{tj}(p_{tj}) \right) \tag{8.92}$$

   sujeto a

$$p_{tj} \quad \in \quad \Pi_j, \qquad \forall t \tag{8.93}$$

   y envía al operador del mercado la producción óptima en cada periodo, $p_{tj}^{(k)}$, $t = 1, \dots, T$, $j = 1, \dots, I$.

3. El operador del mercado calcula el desajuste entre la demanda y la oferta en cada periodo

$$d_t - \sum_{i=1}^{I} p_{ti}^{(k)} \qquad t = 1, \dots, T \tag{8.94}$$

   y actualiza los precios proporcionalmente a este desajuste

$$\lambda_t^{(k+1)} = \lambda_t^{(k)} + K \left(d_t - \sum_{i=1}^{I} p_{ti}^{(k)} \right); \qquad t = 1, \dots, T \tag{8.95}$$

   donde K es una constante de proporcionalidad.

4. Si los precios son casi iguales en dos iteraciones consecutivas, se *para*; la solución óptima ha sido alcanzada (mercado en equilibrio). En caso contrario, se va al paso 2.

8.5.3 Conclusión

El esquema primal es centralizado, y el operador del sistema tiene completa información sobre los costes de producción de cada uno de los productores. Además, tiene la autoridad de decidir la producción de cada uno de ellos.

El esquema dual es descentralizado, cada productor mantiene la privacidad de sus costes de producción y de sus restricciones. Además son los propios productores quienes deciden cuánto producen para maximizar su propio beneficio.

La teoría de la dualidad garantiza que ambos modos de operar conducen a los mismos resultados.

8.6 Condiciones de regularidad

En las secciones anteriores se ha visto el significado y la relevancia de las condiciones de KKT para los PPNL. Estas condiciones son necesarias para la optimalidad en la *mayoría* de los PPNL. Sin embargo, se pueden dar ejemplos sencillos en los que la solución óptima del problema no tiene asociado un punto de KKT. Se han estudiado ciertas condiciones (suficientes) que evitan estas patologías. Estas condiciones reciben el nombre de "condiciones de regularidad" (CR). El estudio de las mismas excede los objetivos de este libro, por lo que aquí simplemente se enuncia alguna de ellas, siendo éstas aplicables a la mayoría de los PPNL que aparecen en las aplicaciones.

- **Condición de Slater**. Esta condición exige que todas las funciones $g_i(\mathbf{x})$ para todo $i \in I(\bar{\mathbf{x}})$ sean convexas y que las funciones $\nabla h_k(\bar{\mathbf{x}})$ para todo $k = 1, \ldots, \ell$ sean linealmente independientes. Además, debe existir un punto $\tilde{\mathbf{x}} \in \mathbb{R}^n$ que cumpla $g_i(\tilde{\mathbf{x}}) < 0$ para todo $i \in I(\bar{\mathbf{x}})$ y $h_k(\tilde{\mathbf{x}}) = 0$ para todo $k = 1, \ldots, \ell$.

- **Condición de independencia lineal**. En esta condición se exige que los gradientes de las restricciones activas sean linealmente independientes, esto es, $\nabla g_i(\bar{\mathbf{x}})$ para todo $i \in I(\bar{\mathbf{x}})$ y $\nabla h_k(\bar{\mathbf{x}})$ para todo $k = 1, \ldots, \ell$ sean linealmente independientes.

Ejemplo 8.10 (casos particulares). En este ejemplo se analizan los siguientes casos:

1. **Problemas no restringidos**. En este caso $S = \mathbb{R}^n$ se cumple la CR y por tanto se puede afirmar en todos los casos que si $\bar{\mathbf{x}}$ es un mínimo local de f y es diferenciable en dicho punto, entonces $\nabla f(\bar{\mathbf{x}}) = 0$.

2. **Problemas con sólo restricciones de igualdad**. Para este caso las anteriores CR conducen a exigir que el conjunto de vectores $\nabla h_k(\bar{\mathbf{x}})$ para $k = 1, \ldots, \ell$ sean linealmente independientes.

3. **Problemas con sólo restricciones de desigualdad**. Las CR para este caso se obtienen eliminando de las CR para el caso general las exigencias para las restricciones de igualdad $\mathbf{h}(\mathbf{x}) = \mathbf{0}$.

■

Ejercicios

8.1 Resuélvanse mediante las condiciones de KKT los siguientes problemas.

(a) Maximizar

$$Z = x_1 - \exp(-x_2)$$

sujeto a

$$\begin{array}{rcl} -\sin x_1 + x_2 & \leq & 0 \\ x_1 & \leq & 3 \end{array}$$

(b) Minimizar

$$Z = (x_1 - 4)^2 + (x_2 - 3)^2$$

sujeto a

$$\begin{array}{rcl} x_1^2 - x_2 & \leq & 0 \\ x_2 & \leq & 4 \end{array}$$

(c) Maximizar

$$Z = x_2$$

sujeto a

$$\begin{array}{rcl} x_1^2 - x_2^2 & \leq & 4 \\ -x_1^2 + x_2 & \leq & 0 \\ x_1,\ x_2 & \geq & 0 \end{array}$$

(d) Minimizar

$$Z = -x_1^3 + x_2$$

sujeto a

$$\begin{array}{rcl} \max\{0, x_1\} - x_2 & \leq & 0 \\ x_1^2 + 3x_2 & \leq & 4 \end{array}$$

(e) Maximizar

$$Z = x_1$$

sujeto a

$$\begin{array}{rcl} x_1^4 + x_1^3 - 3x_1^2 - 5x_1 - x_2 & \geq & 2 \\ x_1 & \leq & 1 \\ x_2 & \geq & 0 \end{array}$$

(f) Minimizar

$$Z = -x_1$$

sujeto a

$$\begin{array}{rcl} x_1^2 + x_2^2 & \leq & 1 \\ (x_1 - 1)^3 - x_2 & \leq & 0 \end{array}$$

(g) Maximizar

$$Z = x_1$$

sujeto a

$$\begin{array}{rcl} x_1^2 - x_2 & \leq & 0 \\ x_1 - x_2 & \leq & -1 \end{array}$$

(h) Minimizar

$$Z = -x_1 + x_2^2$$

sujeto a

$$\begin{array}{rcl} \exp(-x_1) - x_2 & \leq & 0 \\ -\exp(-x_1^2) + x_2 & \leq & 0 \end{array}$$

(i) Minimizar

$$Z = 4x_1^2 + 2x_2^2 + 4x_1x_2 + 3x_1 + \exp(x_1 + 2x_2)$$

sujeto a

$$\begin{array}{rcl} x_1^2 + x_2^2 & \leq & 10 \\ \sqrt{x_2} & \geq & \frac{1}{2} \\ x_2 & \geq & 0 \\ x_1 + 2x_2 & = & 0 \end{array}$$

(j) Maximizar

$$Z = 3x_1 - x_2 + x_3^2$$

sujeto a

$$\begin{array}{rcl} x_1 + x_2 + x_3 & \leq & 0 \\ -x_1 + 2x_2 + x_3^2 & = & 0 \end{array}$$

(k) Minimizar

$$Z = x_1 + x_2$$

sujeto a

$$\begin{array}{rcl} x_1 + x_2^2 & = & 4 \\ -2x_1 - x_2 & \leq & 4 \end{array}$$

(l) Minimizar

$$Z = (x_1 + 1)^2 + (x_2 - 1)^2$$

sujeto a

$$\begin{array}{rcl} (x_1 - 1)(4 - x_1^2 - x_2^2) & \geq & 0 \\ x_2 - \dfrac{1}{2} & = & 0 \\ 100 - 2x_1^2 - x_2^2 & = & 0 \end{array}$$

(m) Maximizar

$$Z = x_2 - x_1$$

sujeto a

$$\begin{array}{rcl} x_1^2 + x_2^2 & \leq & 4 \\ -x_1^2 + x_2 & = & 0 \\ x_1,\ x_2 & \geq & 0 \end{array}$$

8.2 Considérese el problema: Minimizar

$$Z = x_1$$

sujeto a

$$x_1^2 + x_2^2 = 1$$

Calcúlese la función dual y muéstrese que es una función cóncava. Encuéntrese la solución del problema primal y dual comparando los valores de sus funciones objetivo.

8.3 Considérese el problema: Minimizar

$$Z = e^{-x}$$

sujeto a

$$x \geq 0$$

Resuélvase el problema primal. Calcúlese explícitamente la función dual y posteriormente resuélvase el problema dual.

8.4 Considérese el problema: Minimizar

$$Z = 2x_1^2 + x_2^2 - 2x_1x_2 - 6x_2 - 4x_1$$

sujeto a

$$\begin{array}{rcl} x_1^2 + x_2^2 & = & 1 \\ -x_1 + 2x_2 & \leq & 0 \\ x_1 + x_2 & \leq & 8 \\ x_1,\ x_2 & \geq & 0 \end{array}$$

(a) Resuélvase este problema mediante las condiciones de KKT.

(b) Calcúlese explícitamente la función dual.

(c) ¿Existe holgura dual?

8.5 Encuéntrese el mínimo de la función

$$f(x_1, x_2, x_3) = 12(x_1^2 + x_2^2 + x_3^2)$$

sujeto a

$$x_1x_2 + x_1x_3 + x_2x_3 = 1$$

8.6 Encuéntrese la mnima longitud de escalera que puede apoyarse contra una pared si se coloca una caja de dimensiones a y b justo en una esquina de la misma pared.

8.7 Considérese la función

$$\begin{array}{rl} f(y_1, y_2, y_3) = & \min \{y_1x_1 + y_2x_2 + y_3x_3 | x_1 + x_2 + x_3 = 1, \\ & x_1 - x_2 - x_3 = 0, x_i \geq 0\} \end{array}$$

Resuélvase el problema optimización: Maximizar

$$Z = f(y_1, y_2, y_3)$$

sujeto a

$$\begin{array}{rcl} y_1^2 + y_2^2 + y_3^2 & \leq & 1 \\ y_3 - y_2 & \leq & -\dfrac{1}{2} \end{array}$$

8.8 Encuéntrese el máximo de la integral

$$J = \int_{x_1}^{x_2} (e^{-t} - e^{-2t})\, dt$$

respecto a los límites de integración sujeto a la restricción $x_2 - x_1 = c$, donde c es una constante.

8.9 Encuéntrese el máximo y el mínimo de la función

$$f(x_1, x_2) = \int_{x_1}^{x_1+x_2} \log(1 + t + t^2)\, dt$$

sobre la región determinada por las desigualdades

$$\begin{array}{rcl} (x_1 + x_2)^2 & \leq & 4 \\ -2 \;\leq\; x_2 & \leq & 2 \end{array}$$

8.10 Encuéntrese el punto de la superficie $xyz = 1$ más cercano al origen.

8.11 Considérese la función

$$f(x_1, x_2, x_3) = x_1^2 + x_2^2 + x_3^2 + x_1 + x_2 + x_3$$

Muéstrese que f es una función convexa y encuéntrese el máximo y el mínimo de esta función en la región definida por las restricciones

$$\begin{array}{rcl} x_1^2 + x_2^2 + x_3^2 & = & 4 \\ x_3 & \leq & 1 \end{array}$$

8.12 Determínese el máximo y el mínimo de la función

$$f(x_1, x_2, x_3) = x_1^4 x_2 + x_3^4$$

bajo la restricción

$$x_1^4 + x_2^4 + x_3^4 \leq 1$$

8.13 Repítase el ejercicio anterior pero con la función

$$f(x_1, x_2) = x_1^3 - x_1^2 x_2^2 + x_1^2 + x_2^2$$

y las restricciones

$$\begin{array}{rcl} x_1^2 + x_2^2 & \leq & 4 \\ x_2 & \geq & x_1 - 1 \end{array}$$

8.14 Encuéntrese la máxima longitud de una viga que puede pasar por la puerta de una habitación de altura h y situada a una distancia d del techo de la misma. Considérese que la anchura de la puerta no condiciona la solución del problema.

8.15 Considérese el problema: Maximizar

$$x_2 - x_1$$

sujeto a

$$\begin{array}{r} x_1^2 + x_2^2 \leq 4 \\ -x_1^2 + x_2 = 0 \\ x_1, x_2 \geq 0 \end{array}$$

(a) Escríbanse las condiciones de KKT, y encuéntrense los puntos $\mathbf{x}$ que las satisfacen.

(b) Discútase si las condiciones de KKT son necesarias y/o suficientes para la optimalidad de las soluciones de este problema.

8.16 Considérese el problema: Minimizar

$$Z = (x_1 - 3)^2 + (x_2 - 5)^2$$

sujeto a

$$\begin{array}{rrcl} x_1^2 & -x_2 & \leq & 0 \\ x_1 & +2x_2 & \leq & 10 \\ -x_1 & & \leq & 1 \\ x_1, & x_2 & \geq & 0 \end{array}$$

(a) Resuélvase el problema mediante las CKKT.

(b) Sea $X = \{\mathbf{x} \in \mathbb{R}^2 | x_1 + 2x_2 \leq 10\}$. Formúlese el problema dual asociado a este conjunto.

(c) Pruébese que la función dual es diferenciable en el punto $\mathbf{0}$.

(d) Verifíquese que el gradiente en $\mathbf{0}$ es una dirección de ascenso.

(e) Aplíquese el método de mayor pendiente para resolver el problema dual, eligiendo como solución inicial $\mu^0 = \mathbf{0}$.

8.17 Considérese el problema: Minimizar

$$f(\mathbf{x})$$

sujeto a:

$$\mathbf{x} \in X$$
$$\mathbf{A}\mathbf{x} = \mathbf{b}$$

donde X es un politopo, y f es una función cóncava.

(a) Escríbase la función dual asociada al conjunto X.

(b) Pruébese que existe un punto extremo de X que minimiza la función lagrangiana.

(c) Muéstrese que la función dual es cóncava y definida por tramos.

8.18 Antonio escribe libros que posteriormente Sebastián edita y vende. Supóngase que el número de copias vendidas depende solamente del precio de venta. Sebastián y Antonio han calculado por separado el precio de venta y han comprobado que estos precios difieren en la cantidad

$$\frac{c}{2(1-\gamma)}$$

donde c es el coste de producción del libro y γ la cantidad que se le paga al autor (Antonio) por cada libro vendido. ¿Han calculado correctamente los precios más convenientes de venta del libro? ¿Por qué son diferentes?

Capítulo 9

Métodos computacionales para programación no lineal

En este capítulo se desarrollan algoritmos para calcular las soluciones de los PPNL. En general, estos métodos generan una sucesión de puntos cuyo límite es una solución del problema bajo estudio. Para asegurar la convergencia, se debe suponer que el PPNL es un problema convexo diferenciable. No obstante, en la práctica, estos algoritmos son aplicables incluso cuando no se satisfacen estas condiciones de convergencia.

El criterio de parada se basa, usualmente, en las condiciones de optimalidad de KKT. Cuando un punto de la sucesión generada las satisface con una cierta tolerancia, el procedimiento se detiene y el punto correspondiente se considera como un mínimo local.

En algunas ocasiones, los problemas de optimización con restricciones se resuelven convirtiéndolos en otros equivalentes sin restricciones. Estos métodos de optimización con restricciones se desarrollan sobre la base de que existen algoritmos para la resolución de los problemas sin restricciones que, formalmente hablando, son un caso especial de los problemas con restricciones donde el dominio de definición es $S = \mathbb{R}^n$.

En la sección 9.1 de este capítulo se abordan los métodos para problemas sin restricciones, y posteriormente, en la sección 9.2, el caso más general de problemas con restricciones.

Se comienza la discusión de los métodos sin restricciones con el problema de minimizar una función de una sola variable. Seguidamente, en la sección 9.1.2, se estudian los métodos de direcciones de descenso para minimizar funciones de varias variables. Estos métodos transforman el problema de minimizar una función de varias variables en una secuencia de problemas unidimensionales.

En la sección 9.2 se discuten varias estrategias para resolver los problemas con restricciones de igualdad y desigualdad. La primera evita estas restricciones (o las simplifica) mediante la resolución del problema dual. La segunda estrategia, que conduce a los *métodos de penalización y de barrera*, convier-

te el problema con restricciones en una sucesión equivalente de problemas de optimización sin restricciones.

Es importante mencionar que el software GAMS permite formular fácilmente los problemas de optimización sin necesidad de conocer los métodos de resolución.

9.1 Algoritmos de optimización para problemas sin restricciones

9.1.1 Métodos de búsqueda lineal

Se comienza con el problema de optimización unidimensional: Minimizar

$$Z = f(x) \tag{9.1}$$

donde $x \in \mathbb{R}$ y $f : \mathbb{R} \to \mathbb{R}$ es una función diferenciable para todo $x \in \mathbb{R}$.

Existe una gran cantidad de procedimientos para resolver este problema. Éstos se clasifican en dos categorías:

1. Los que emplean información sobre las derivadas. Estos métodos operan directamente sobre la condición necesaria de optimalidad (condición KKT)

$$f'(x^*) = 0 \tag{9.2}$$

   para obtener x^*, que es la solución buscada.

2. Los que emplean solamente evaluaciones de la función objetivo. Estos algoritmos interpolan iterativamente la función objetivo y derivan fórmulas para calcular el mínimo de la función interpolada. Estos mínimos generan una sucesión que converge a la solución buscada.

Búsquedas lineales con derivadas

Nótese que si la función f es convexa, entonces el punto x^* es un mínimo global del problema unidimensional si y sólo si se cumple la ecuación (9.2); entonces los problemas (9.1) y (9.2) son equivalentes. Se trata de calcular la raíz de la ecuación (9.2), pudiendo emplearse métodos para el cálculo de raíces, tales como el método de Newton, métodos quasi-Newton, o el método de la bisección. En esta sección se discute los dos primeros.

Primeramente se recordará al lector cómo se emplean estos métodos para resolver la ecuación $g(x) = 0$. Estos métodos generan una sucesión que, en caso de converger, lo hace a alguna raíz de la ecuación anterior. El modo de obtener estas sucesiones es el siguiente.

Método de Newton. El valor de la función $g(x)$ en el punto $x^{(t+1)}$ se puede aproximar por

$$g(x^{(t+1)}) = g(x^{(t)}) + g'(x^{(t)})(x^{(t+1)} - x^{(t)})$$

Como se quiere que esta expresión valga cero, debe ser

$$x^{(t+1)} = x^{(t)} - \frac{g(x^{(t)})}{g'(x^{(t)})}$$

Método quasi-Newton o de la secante. Sustituyendo en la expresión anterior $g'(x^{(t)})$por la aproximación

$$\frac{g(x^{(t)}) - g(x^{(t-1)})}{x^{(t)} - x^{(t-1)}}$$

se llega a

$$x^{(t+1)} = x^{(t)} - \frac{g(x^{(t)})}{g(x^{(t)}) - g(x^{(t-1)})}(x^{(t)} - x^{(t-1)})$$

Volviendo a la resolución del problema 9.1 y aplicando las fórmulas anteriores a la ecuación (9.2), se obtiene

1. **Método de Newton**

$$x^{(t+1)} = x^{(t)} - \frac{f'(x^{(t)})}{f''(x^{(t)})}$$

   que emplea la primera y segunda derivada de la función objetivo.

2. **Método quasi-Newton o de la secante**

$$x^{(t+1)} = x^{(t)} - \frac{f'(x^{(t)})}{f'(x^{(t)}) - f'(x^{(t-1)})}(x^{(t)} - x^{(t-1)})$$

   que requiere usar únicamente la primera derivada de la función objetivo.

Ejemplo 9.1 (búsquedas lineales con derivadas). Considérese el problema: Minimizar

$$Z = f(x) = (x-1)^4 \tag{9.3}$$

Es claro que la solución óptima se alcanza en $x^* = 1$. Se ilustra en este ejemplo el método de Newton. La primera y la segunda derivadas de la función anterior son

$$\begin{array}{rcl} f'(x) & = & 4(x-1)^3 \\ f''(x) & = & 12(x-1)^2 \end{array}$$

y la sucesión generada por el método de Newton es

$$\begin{array}{rcl} x^{(t+1)} & = & x^{(t)} - \dfrac{f'(x^{(t)})}{f''(x^{(t)})} = x^{(t)} - \dfrac{4(x^{(t)}-1)^3}{12(x^{(t)}-1)^2} \\ & = & x^{(t)} - \dfrac{1}{3}(x^{(t)}-1) = \dfrac{2}{3}x^{(t)} + \dfrac{1}{3} \end{array} \tag{9.4}$$

Tabla 9.1: Resultados de los métodos de Newton y quasi-Newton.

	Método de Newton		Método de quasi-Newton	
t	$x^{(t)}$	$f'(x^{(t)})$	$x^{(t)}$	$f'(x^{(t)})$
1	0.000	-4.000	0.000	-4.000
2	0.333	-1.185	0.333	-1.185
3	0.556	-0.351	0.474	-0.583
4	0.704	-0.104	0.610	-0.238
5	0.803	-0.031	0.703	-0.104
6	0.868	-0.009	0.777	-0.045
7	0.912	-0.003	0.831	-0.019
8	0.942	-8.02×10^{-4}	0.873	-0.008
9	0.961	-2.38×10^{-4}	0.904	-0.003
10	0.974	-7.04×10^{-5}	0.927	-0.002

Nótese que si la sucesión converge a un punto $\bar{x}$, entonces, tomando límites a ambos lados de la expresión (9.4), se obtiene $\bar{x} = \frac{2}{3}\bar{x} + \frac{1}{3}$, y esta ecuación tiene como única solución $\bar{x} = x^* = 1$. Esto es, si el método converge lo hace a la solución óptima.

Si se aplícase el método quasi-Newton, se obtendría

$$\begin{aligned} x^{(t+1)} &= x^{(t)} - \frac{f'(x^{(t)})}{\dfrac{f'(x^{(t)}) - f'(x^{(t-1)})}{x^{(t)} - x^{(t-1)}}} \\ &= x^{(t)} - \left[\frac{x^{(t)} - x^{(t-1)}}{(x^{(t)} - 1))^3 - (x^{(t-1)} - 1)^3}\right](x^{(t)} - 1)^3 \qquad (9.5) \\ &= x^{(t)} - \frac{x^{(t)} - x^{(t-1)}}{1 - \left(\dfrac{x^{(t-1)} - 1}{x^{(t)} - 1}\right)^3} \end{aligned}$$

Obsérvese que el método de Newton requiere un único valor inicial mientras que el método quasi-Newton emplea dos valores iniciales. La tabla 9.1 muestra las primeras diez iteraciones de ambos métodos. Los puntos iniciales elegidos son: $x_1 = 0$ para el de Newton, y $x_1 = 0$ y $x_2 = \frac{1}{3}$ para el de quasi-Newton.

■

Búsquedas lineales sin derivadas

En esta sección se analiza el método de búsqueda lineal mediante interpolación cuadrática. Este método interpola la función objetivo mediante una parábola, lo que requiere conocer tres puntos de la gráfica de la función f, y considera que este polinomio cuadrático es una aproximación de la función objetivo. Por

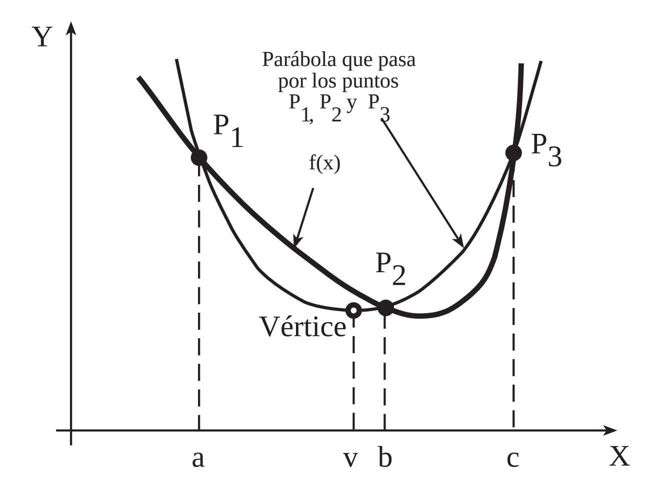


Figura 9.1: Función objetivo a minimizar junto a una parábola interpolante.

tanto, el método toma el vértice de la parábola como una aproximación del mínimo de la función objetivo. El proceso se repite con el vértice hallado y con dos de los tres puntos iniciales de interpolación. Más formalmente, se supone que se minimiza la función convexa $f(x)$, y se consideran tres puntos $a < b < c$ tales que se cumple (véase la figura 9.1)

$$f(a) \geq f(b) \leq f(c) \tag{9.6}$$

Se puede suponer que al menos una de las dos desigualdades anteriores se cumple estrictamente. En caso contrario, por convexidad, f sería constante y todos los puntos del intervalo $[a, c]$ serían mínimos. Seguidamente, se ajusta una parábola que pase por los puntos $(a, f(a)), (b, f(b))$ y $(c, f(c))$. Su vértice viene dado por la expresión

$$v = b - \frac{1}{2} \frac{(b-a)^2[f(b) - f(c)]^2 - (b-c)^2[f(b) - f(a)]}{(b-a)[f(b) - f(c)] - (b-c)[f(b) - f(a)]} \tag{9.7}$$

Si los tres puntos fuesen colineales, el denominador sería cero y la expresión anterior no estaría definida, pero esta situación es imposible bajo la suposición de $f(a) > f(b)$ ó $f(b) < f(c)$. Nótese que la condición (9.6) garantiza que el vértice de la parábola es un mínimo (y no un máximo), y que está situado en el intervalo (a, c).

Se dispone pues de un conjunto de cuatro puntos $\{a, b, c, v\}$ para elegir un nuevo conjunto de tres puntos con los que se pueda repetir el proceso. La selección debe garantizar que se satisfaga la condición (9.6). Pueden aparecer tres casos:

Caso 1. $v < b$ (véase la figura 9.2). Hay dos posibilidades: (a) si $f(b) \geq f(v)$ el mínimo de f se encuentra en el intervalo (a, b); entonces, el nuevo conjunto

Tabla 9.2: Resumen de los cálculos para búsquedas lineales mediante interpolación cuadrática.

t	a	b	c	v	$f(v)$	$f(b)$
1	0.0000	0.3333	1.500	0.9545	4.268×10^{-6}	0.1975
2	0.3333	0.9545	1.500	1.0720	2.801×10^{-5}	4.268×10^{-6}
3	0.3333	0.9545	1.072	1.0130	3.240×10^{-8}	4.268×10^{-6}
4	0.9545	1.0130	1.072	0.9918	4.509×10^{-9}	3.240×10^{-8}
5	0.9545	0.9918	1.013	1.0020	2.715×10^{-11}	4.509×10^{-9}
6	0.9918	1.0020	1.013	0.9984	6.077×10^{-12}	2.715×10^{-11}
7	0.9918	0.9984	1.002	1.0000	9.785×10^{-15}	6.077×10^{-12}
8	0.9984	1.0000	1.002	0.9997	4.799×10^{-15}	9.785×10^{-15}
9	0.9997	1.0000	1.002	1.0000	3.821×10^{-19}	9.785×10^{-15}
10	0.9997	1.0000	1.000	1.0000	3.193×10^{-19}	3.821×10^{-19}

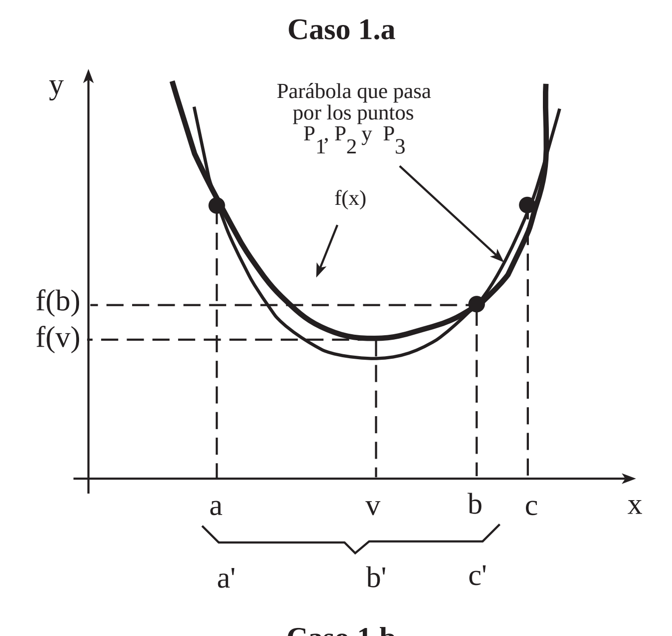
de tres puntos es $\{a', b', c'\} = \{a, v, b\}$; (b) por otro lado, si $f(b) < f(v)$, el mínimo de f está en el intervalo (v, c); y el nuevo conjunto de tres puntos es $\{a', b', c'\} = \{v, b, c\}$.

Caso 2. $v > b$. Por un argumento similar al caso 1, si $f(v) \geq f(b)$, se toman $\{a', b', c'\} = \{a, b, v\}$, y si $f(v) < f(b)$, se hace $\{a', b', c'\} = \{b, v, c\}$.

Caso 3. $v = b$. En este caso se obtienen los tres puntos iniciales y es imposible decidir en qué intervalo reducido se encuentra la solución. Para evitar este problema, se reemplaza b por $(a+b)/2$ y se repite el proceso, con la garantía de que en el siguiente intento se estará en el caso 1 o en el 2.

La técnica de interpolación cuadrática, descrita anteriormente, tiene asegurada la convergencia bajo la hipótesis de que f es una función convexa. Una modificación que simplifica la codificación del procedimiento consiste en sistemáticamente reemplazar el vértice v por el último punto generado. Este nuevo procedimiento presenta una convergencia local, esto es, la convergencia depende de lo cerca que estén los puntos iniciales de la solución óptima.

Ejemplo 9.2 (ejemplo de búsquedas lineales mediante interpolación cuadrática). En este apartado se ilustra el método de búsqueda lineal mediante interpolación cuadrática sobre el ejemplo 9.1. La tabla 9.2 muestra los diferentes conjuntos de tres puntos que se van generando en función de los valores $v, b, f(v)$ y $f(b)$. Nótese que para este ejemplo la convergencia del método de interpolación cuadrática es superior a la del método de Newton y a la del método quasi-Newton. Esto ocurre porque la solución x^* no es una raíz simple de $f'(x)$ y el método de Newton presenta una convergencia lineal. Para los problemas en los que la solución es una raíz simple, el método de Newton tiene una convergencia cuadrática, siendo el método más rápido.



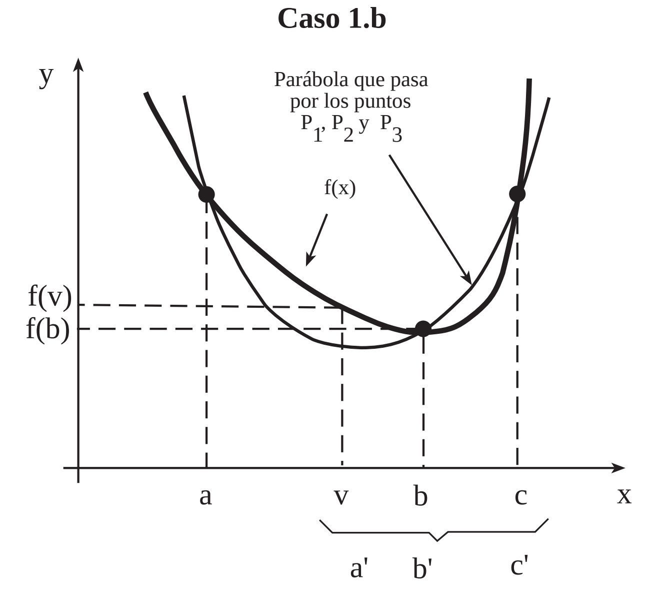


Figura 9.2: Ilustración de los casos 1a y 1b de la búsqueda lineal mediante interpolación cuadrática.

∎

9.1.2 Optimización sin restricciones

En esta sección se considera el problema de minimizar una función de varias variables empleando derivadas. Considérese el problema: Minimizar

$$Z = f(\mathbf{x}) \tag{9.8}$$

donde $\mathbf{x} \in \mathbb{R}^n$ y $f : \mathbb{R}^n \to \mathbb{R}$ es una función diferenciable en todo $\mathbf{x} \in \mathbb{R}^n$. Este problema se puede resolver aplicando los llamados *métodos de descenso*. Estos métodos generan una secuencia de puntos que van reduciendo el valor de la función objetivo, o más formalmente generan una sucesión $\{\mathbf{x}^{(t)}\}$ que cumple $f(\mathbf{x}^{(1)}) > f(\mathbf{x}^{(2)}) > \cdots > f(\mathbf{x}^{(t)}) > \cdots$. Estos procedimientos terminan cuando se satisface algún criterio de parada. La familia más importante de estos métodos son los llamados *métodos de direcciones de descenso*. Una iteración de estos algoritmos consta de dos etapas fundamentales:

1. **Generación de la dirección de descenso**. En esta etapa, dado un punto $\mathbf{x}^{(t)}$, se obtiene una dirección de búsqueda $\mathbf{d}^{(t)}$ de modo que un ligero movimiento desde el punto $\mathbf{x}^{(t)}$ en dicha dirección hace decrecer el valor de la función objetivo. Más formalmente, se supone que existe un número positivo $\bar{\alpha}$ que cumple

$$f(\mathbf{x} + \alpha \mathbf{d}) < f(\mathbf{x}) \text{ para todo } \alpha \in (0, \bar{\alpha})$$

   Se dice en este caso que la dirección $\mathbf{d}$ es de descenso para la función f en el punto $\mathbf{x}$.

2. **Búsqueda lineal**. Habiendo obtenido la dirección de descenso $\mathbf{d}^{(t)}$ de f en el punto $\mathbf{x}^{(t)}$, se plantea el problema de cuánto hay que desplazarse en esa dirección, dicho avance se denomina longitud de paso y se denota por α_t. El nuevo punto de la sucesión es $\mathbf{x}^{(t+1)} = \mathbf{x}^{(t)} + \alpha_t \mathbf{d}^{(t)}$ y éste tiene un valor de la función objetivo menor que el valor en el punto original $\mathbf{x}^{(t)}$. En muchas ocasiones, la longitud de paso α_t se calcula para obtener el mínimo de la función objetivo a lo largo de la dirección $\mathbf{d}^{(t)}$.

El lema siguiente caracteriza las direcciones de descenso mediante el vector gradiente $\nabla f(\mathbf{x})$.

Lema 9.1 *Sea* $f : \mathbb{R}^n \to \mathbb{R}$ *diferenciable en* $\mathbf{x} \in \mathbb{R}^n$*. Sea* $\mathbf{d}$ *un vector de* $\mathbb{R}^n$*. Si* $\nabla f(\mathbf{x})^T \mathbf{d} < 0$*, entonces* $\mathbf{d}$ *es una dirección de descenso de* f *en* $\mathbf{x}$*.*

Demostración. De la diferenciabilidad de f en $\mathbf{x}$, se obtiene que

$$f'(\mathbf{x}; \mathbf{d}) = \lim_{\alpha \to 0^+} \frac{f(\mathbf{x} + \alpha \mathbf{d}) - f(\mathbf{x})}{\alpha} = \nabla f(\mathbf{x})^T \mathbf{d} < 0.$$

Entonces, existe un número positivo $\bar{\alpha}$ que cumple

$$\frac{f(\mathbf{x} + \alpha \mathbf{d}) - f(\mathbf{x})}{\alpha} < 0 \text{ para todo } \alpha \in (0, \bar{\alpha})$$

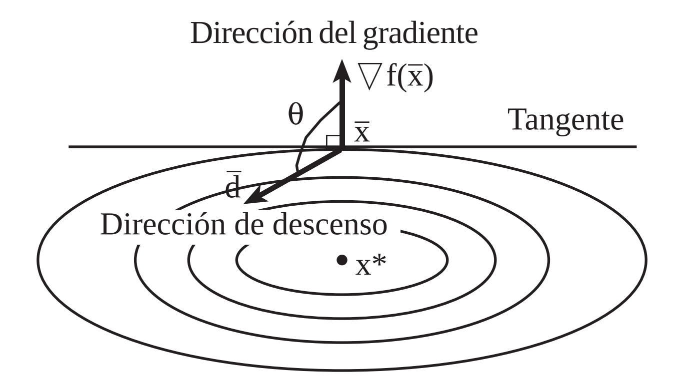


Figura 9.3: Ilustración de las direcciones de descenso en $\mathbb{R}^2$.

y por tanto $\mathbf{d}$ es una dirección de descenso de f en $\mathbf{x}$. ■

La figura 9.3 muestra que una dirección de descenso tiene una proyección positiva en la dirección del vector gradiente cambiado de signo.

Los algoritmos de direcciones de descenso tienen la siguiente estructura.

Algoritmo 9.1 (algoritmos de direcciones de descenso).

Paso 1. (iniciación). *Se elige un punto inicial* $\mathbf{x}^{(1)} \in \mathbb{R}^n$, *y se toma* $t = 1$.

Paso 2. (generación de la dirección de búsqueda). *Se obtiene una dirección de descenso,* $\mathbf{d}^{(t)}$, *de* f *en el punto* $\mathbf{x}^{(t)}$.

Paso 3. (comprobación de la optimalidad). *Si* $\mathbf{d}^{(t)} = \mathbf{0}$, *entonces se para (*$\mathbf{x}^{(t)}$ *es un punto de KTT). En caso contrario, se continúa.*

Paso 4. (búsqueda lineal). *Se busca la longitud de paso,* α_t, *mediante la resolución del problema unidimensional: Minimizar*

$$Z_{LS} = f(\mathbf{x}^{(t)} + \alpha \mathbf{d}^{(t)})$$

sujeto a

$$\alpha \geq 0$$

Paso 5. (actualización). *Se toma* $\mathbf{x}^{(t+1)} = \mathbf{x}^{(t)} + \alpha_t \mathbf{d}^{(t)}$.

Paso 5. (criterio de parada). *Si* $\|\mathbf{x}^{(t+1)} - \mathbf{x}^{(t)}\| \leq \epsilon$, *entonces se para. Si no es así, se toma* $t = t + 1$, *y se va al paso 2.*

■

Los métodos anteriores realizan una búsqueda de un mínimo en un espacio n-dimensional realizando búsquedas unidimensionales en el conjunto de direcciones $\{\mathbf{d}^{(t)}\}$. Los métodos de búsquedas lineales de Newton y quasi-Newton

emplean la primera y segunda (sólo el de Newton) derivada de la función $g(\alpha) = f(\mathbf{x}^{(t)} + \alpha\mathbf{d}^{(t)})$. Éstas se calculan como

$$\begin{aligned} g'(\alpha) &= \nabla f(\mathbf{x}^{(t)} + \alpha\mathbf{d}^{(t)})^T\mathbf{d}^{(t)} \\ g''(\alpha) &= (\mathbf{d}^{(t)})^T\nabla^2 f(\mathbf{x}^{(t)} + \alpha\mathbf{d}^{(t)})\mathbf{d}^{(t)} \end{aligned} \tag{9.9}$$

La eficiencia de los métodos de direcciones de descenso depende críticamente de la eficiencia en realizar la búsqueda unidimensional. En la práctica, el cálculo de la longitud de paso α_t es un compromiso entre el coste de cálculo y la reducción conseguida en el valor de la función objetivo f. Algunas estrategias prácticas consisten en realizar *búsquedas lineales inexactas* para determinar α_t, de modo que se reduzca adecuadamente la función f y se requiera poco tiempo en su cálculo. El objetivo de estos procedimientos es mantener la convergencia del procedimiento a un punto estacionario pero reduciendo el número de evaluaciones, tanto de la función objetivo como de su vector gradiente. Un método habitual de búsqueda inexacta es la llamada *regla de Armijo.* Sean $0 < \varepsilon < 1$ y $\delta > 1$, dos parámetros que controlan, la longitud de paso para que no sea ni demasiado grande ni demasiado pequeña, respectivamente. La aproximación de Taylor de primer orden de g en 0 con un incremento α es $g(0) + \alpha g'(0)$. Si se perturba la recta anterior y se define

$$\bar{g}(\alpha) = g(0) + \varepsilon\alpha g'(0) = f(\mathbf{x}^{(t)}) + \varepsilon\alpha\nabla f(\mathbf{x}^{(t)})^T\mathbf{d}^{(t)}$$

Una longitud de paso α_t se acepta si se cumple las dos condiciones siguientes:

$$g(\alpha_t) \leq \bar{g}(\alpha_t) \tag{9.10}$$

$$g(\delta\alpha_t) > \bar{g}(\delta\alpha_t) \tag{9.11}$$

La figura 9.4 ilustra la regla de Armijo. La condición (9.10) garantiza un descenso suficiente del valor de la función g. Esta condición no es suficiente por ella misma para asegurar que el algoritmo de lugar a un progreso convergente. La figura 9.4 muestra que para todo valor de α suficientemente pequeño, se cumple esta condición. Por otro lado, la condición (9.11) garantiza un avance mínimo en dicha dirección.

En su forma básica la regla de Armijo sigue los pasos siguientes.

Algoritmo 9.2 (regla de Armijo).

Paso 1(iniciación). Elegir $\hat{\alpha} > 0$, $\varepsilon \in (0,1)$ y $\delta > 1$. Valores típicos son $\varepsilon = 0.2$ y $\delta = 2$ ó $\delta = 10$. Se toma $\alpha = \hat{\alpha}$.

Paso 2. Si $g(\alpha) \leq \bar{g}(\alpha)$, entonces se va al paso 3. En caso contrario se va al paso 4.

Paso 3. Si $g(\delta\alpha) > \bar{g}(\delta\alpha)$, entonces se *para*, y se toma $\alpha_t = \alpha$. En caso contrario, se toma $\alpha = \delta\alpha$ y se va al paso 2.

Paso 4. Si $g(\alpha/\delta) \leq \bar{g}(\alpha/\delta)$, entonces se para. En caso contrario, se toma $\alpha = \alpha/\delta$ y se va al paso 3. ■

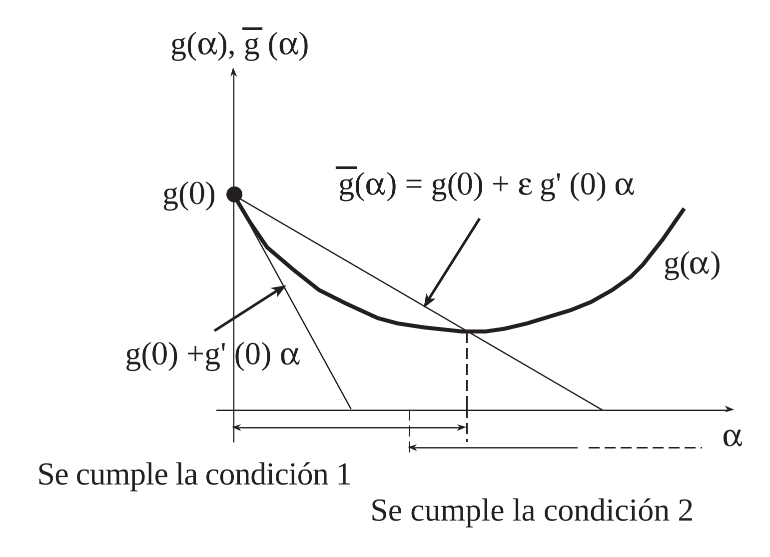


Figura 9.4: Ilustración gráfica de la regla de Armijo.

Esta estrategia es adecuada para ciertos métodos de direcciones de descenso, como el método de Newton pero no resulta tan eficiente para los llamados métodos quasi-Newton y para los métodos de direcciones conjugadas. Estos métodos se verán mas adelante. El lector interesado puede encontrar un análisis detallado de este tema en Nocedal y Wright [81].

Los métodos de direcciones de descenso convergen a puntos candidatos a óptimos locales, y paran cuando el punto actual está lo suficientemente próximo a uno de estos candidatos. La regla de parada más utilizada es $\|\nabla f(\mathbf{x}^{(t)})\| \leq \varepsilon$, que está basada en la condición necesaria de optimalidad $\nabla f(\mathbf{x}) = \mathbf{0}$ y en la continuidad de la función $\nabla f(\mathbf{x})$ que garantiza que la norma del gradiente es pequeña en un entorno del óptimo.

Como ya se ha mencionado, los métodos de direcciones de descenso constan de dos pasos fundamentales. En el primero se selecciona una dirección de descenso y en el siguiente se calcula el desplazamiento a lo largo de esa dirección. Al variar el modo de obtener la dirección de descenso se obtiene un método diferente. Los más usuales son:

1. **Método del gradiente**. Este algoritmo emplea como dirección de descenso en el punto $\mathbf{x}^{(t)}$ la dirección $\mathbf{d}^{(t)} = -\nabla f(\mathbf{x}^{(t)})$, que, empleando el lema 9.1, se puede mostrar que es de descenso. Veámoslo. Si $\nabla f(\mathbf{x}^{(t)}) \neq \mathbf{0}$ se cumple la siguiente expresión

$$\nabla f(\mathbf{x}^{(t)})^T \mathbf{d}^{(t)} = -\nabla f(\mathbf{x}^{(t)})^T \nabla f(\mathbf{x}^{(t)}) = -\left(\|\nabla f(\mathbf{x}^{(t)})\|\right)^2 < 0 \quad (9.12)$$

Entonces, $\mathbf{d}^{(t)}$ es una dirección de descenso de f en $\mathbf{x}^{(t)}$.

2. **Método de Newton**. Este algoritmo elige como dirección de búsqueda

$$\mathbf{d}^{(t)} = -\left[\nabla^2 f(\mathbf{x}^{(t)})\right]^{-1} \nabla f(\mathbf{x}^{(t)})$$

Es importante notar que la dirección $\mathbf{d}^{(t)}$ no se puede calcular si $\nabla^2 f(\mathbf{x}^{(t)})$ es una matriz singular. Incluso en el caso de que no lo fuese, $\mathbf{d}^{(t)}$ no sería necesariamente una dirección de descenso cuando la matriz $\nabla^2 f(\mathbf{x}^{(t)})$ no es definida positiva. Supóngase que $\nabla^2 f(\mathbf{x}^{(t)})$ es definida positiva y que el vector $\nabla f(\mathbf{x}^{(t)}) \neq \mathbf{0}$, entonces su matriz inversa también es definida positiva:

$$\nabla f(\mathbf{x}^{(t)})^T \mathbf{d}^{(t)} = \nabla f(\mathbf{x}^{(t)})^T \left(-\left[\nabla^2 f(\mathbf{x}^{(t)})\right]^{-1} \right) \nabla f(\mathbf{x}^{(t)}) < 0 \qquad (9.13)$$

y empleando el lema 9.1, $\mathbf{d}^{(t)}$ es una dirección de descenso de f en $\mathbf{x}^{(t)}$. La programación convexa es un caso importante para el que la dirección de Newton es de descenso. Empleando la caracterización dada en el teorema 8.4, se tiene que para las funciones convexas la matriz hessiana $\nabla^2 f(\mathbf{x}^{(t)})$ es semidefinida positiva y si se exige que sea no singular, para poder calcular la dirección, se obtiene que es una matriz definida positiva.

3. **Métodos quasi-Newton**. La dirección de búsqueda en el método de Newton requiere que la matriz hessiana sea invertible. Además se necesita un gran esfuerzo computacional para obtener dicha matriz, incluso en problemas de un tamaño moderado. Para evitar estas dos dificultades los métodos quasi-Newton aproximan la matriz $[\nabla^2 f(\mathbf{x}^{(t)})]$ por una matriz definida positiva $\mathbf{B}^{(t)}$, que se actualiza sucesivamente y que converge a la verdadera matriz hessiana. La dirección de búsqueda se calcula mediante la expresión

$$\mathbf{d}^{(t)} = -\left[\mathbf{B}^{(t)}\right]^{-1} \nabla f(\mathbf{x}^{(t)})$$

Un ejemplo de estos algoritmos es el llamado método de *Davidon–Fletcher–Powell* (DFP). Con el objeto de describir el método, se denomina

$$\mathbf{H}^{(t)} = \left[\mathbf{B}^{(t)}\right]^{-1}$$

La fórmula de actualización de las aproximaciones de la inversa de la matriz hessiana es

$$\mathbf{H}^{(t+1)} = \mathbf{H}^{(t)} + \frac{\mathbf{p}^{(t)}\left(\mathbf{p}^{(t)}\right)^T}{\left(\mathbf{p}^{(t)}\right)^T \mathbf{q}^{(t)}} - \frac{\mathbf{H}^{(t)}\mathbf{q}^{(t)}\left(\mathbf{q}^{(t)}\right)^T \mathbf{H}^{(t)}}{\left(\mathbf{q}^{(t)}\right)^T \mathbf{H}^{(t)} \mathbf{q}^{(t)}} \qquad (9.14)$$

donde $\mathbf{p}^{(t)} = \mathbf{x}^{(t+1)} - \mathbf{x}^{(t)}$, $\mathbf{q}^{(t)} = \nabla f(\mathbf{x}^{(t+1)}) - \nabla f(\mathbf{x}^{(t)})$, y $\mathbf{H}^{(1)}$ es la matriz identidad $\mathbf{I}_n$. Se puede probar que la matriz $\mathbf{H}^{(t+1)}$ aproxima iterativamente la inversa de la matriz hessiana (véase Luenberger [68]).

Esta fórmula fué primero desarrollada por Davidon [31] y posteriormente mejorada por Fletcher y Powell [39]. Otro ejemplo es el método de *Broyden–Goldfarb–Shanno* (BFGS) que emplea la actualización:

$$\mathbf{H}^{(t+1)} = \left(\mathbf{I}_n - \frac{\mathbf{p}^{(t)}(\mathbf{q}^{(t)})^T}{(\mathbf{q}^{(t)})^T\mathbf{p}^{(t)}}\right)\mathbf{H}^{(t)}\left(\mathbf{I}_n - \frac{\mathbf{q}^{(t)}(\mathbf{p}^{(t)})^T}{(\mathbf{q}^{(t)})^T\mathbf{p}^{(t)}}\right) + \frac{\mathbf{p}^{(t)}(\mathbf{p}^{(t)})^T}{(\mathbf{q}^{(t)})^T\mathbf{p}^{(t)}}. \quad (9.15)$$

4. **Caso general**. Los métodos anteriores son casos particulares del caso general en el que se elige como dirección de búsqueda la transformación lineal del gradiente cambiado de signo mediante una matriz definida positiva. Sea $\mathbf{A}^{(t)}$ una matriz definida positiva, por tanto , $\mathbf{x}^T\mathbf{A}^{(t)}\mathbf{x} > 0$ para todo $\mathbf{x} \neq \mathbf{0}$. Sea $\mathbf{d}^{(t)} = \mathbf{A}^{(t)}[-\nabla f(\mathbf{x}^{(t)})]$, entonces

$$\nabla f(\mathbf{x}^{(t)})^T\mathbf{d}^{(t)} = \nabla f(\mathbf{x}^{(t)})^T\mathbf{A}^{(t)}[-f(\mathbf{x}^{(t)})] < 0$$

y empleando el lema 9.1, $\mathbf{d}^{(t)}$ es una dirección de descenso. Un importante conjunto de matrices definidas positivas es el de las *matrices de proyección*. Una matriz $n \times n$ se dice que es una *proyección* si $\mathbf{P}^T = \mathbf{P}$ y $\mathbf{PP} = \mathbf{P}$.

5. **Métodos de direcciones conjugadas**. Este método calcula la dirección de búsqueda como

$$\mathbf{d}^{(t)} = -\nabla f(\mathbf{x}^{(t)}) + \beta_{t-1}\mathbf{d}^{(t-1)} \quad (9.16)$$

donde $\mathbf{d}^{(1)} = \mathbf{0}$. Empleando el lema 9.1 para probar que es de descenso, se obtiene

$$\nabla f(\mathbf{x}^{(t)})^T\mathbf{d}^{(t)} = -\|\nabla f(\mathbf{x}^{(t)})\|^2 + \beta_{t-1}\nabla f(\mathbf{x}^{(t)})^T\mathbf{d}^{(t-1)}. \quad (9.17)$$

Nótese que $\mathbf{x}^{(t)} = \mathbf{x}^{(t-1)} + \alpha_t\mathbf{d}^{(t-1)}$ minimiza la función f a lo largo del conjunto $x^{(t-1)} + \alpha\mathbf{d}^{(t-1)}$. Entonces la función $g(\alpha) = f(x^{(t-1)} + \alpha\mathbf{d}^{(t-1)})$ alcanza un mínimo en el valor $\alpha = \alpha_{t-1}$ y $g'(\alpha_{t-1}) = 0$. Empleando (9.9) (reemplazando t por $t-1$), se obtiene

$$g'(\alpha_{t-1}) = \nabla f(x^{(t-1)} + \alpha_{t-1}\mathbf{d}^{(t-1)})^T\mathbf{d}^{(t-1)} = \nabla f(x^{(t)})^T\mathbf{d}^{(t-1)} = 0$$

Como resultado, el segundo término de (9.17) es cero, y $\nabla f(\mathbf{x}^{(t)})^T\mathbf{d}^{(t)} < 0$ si $\nabla f(\mathbf{x}^{(t)}) \neq \mathbf{0}$. Hay muchas variantes de los métodos de direcciones conjugadas que difieren en la manera de elegir el parámetro β_{t-1}. El método de Fletcher–Reeves emplea la fórmula

$$\beta_{t-1}^{\mathrm{FR}} = \begin{cases} 0 & \text{si } t \text{ es un múltiplo de } n \\ \dfrac{\|\nabla f(\mathbf{x}^{(t)})\|^2}{\|\nabla f(\mathbf{x}^{(t-1)})\|^2} & \text{en caso contrario} \end{cases}$$

El método de Polak–Ribière define este parámetro como:

$$\beta_{t-1}^{\mathrm{FR}} = \begin{cases} 0 & \text{si } t \text{ es un múltiplo de } n \\ \dfrac{\nabla f(\mathbf{x}^{(t)})^T\left(\nabla f(\mathbf{x}^{(t)}) - \nabla f(\mathbf{x}^{(t-1)})\right)}{\|\nabla f(\mathbf{x}^{(t-1)})\|^2} & \text{en caso contrario} \end{cases}$$

Estos métodos *reinician* el procedimiento cada n iteraciones, tomando $\beta_{t-1} = 0$; esto es, emplean una iteración del método del gradiente. La motivación de está reiniciación se basa en que para funciones cuadráticas, se alcanza el óptimo después de realizar n pasos.

Estos métodos, como ya se ha comentado, tienen convergencia finita (tras n iteraciones) para las funciones convexas cuadráticas y asumiendo que las búsquedas lineales se realizan de forma exacta. Además, para este caso, los dos métodos anteriores de direcciones conjugadas son idénticos. Ciertos experimentos numéricos muestran que el método de Polak–Ribière es el más robusto y eficiente de los dos.

Ejemplo 9.3 (ejemplo numérico). Para ilustrar los métodos anteriores, considérese el problema: Minimizar

$$f(x_1, x_2) = -2x_1x_2 - 2x_2 + x_1^2 + 2x_2^2$$

cuya solución óptima es $(1, 1)^T$.

1. Método del gradiente. Se realizan dos iteraciones del algoritmo comenzando en el punto $\mathbf{x}^{(1)} = \mathbf{0}$. En la iteración t, este método emplea como dirección de búsqueda la opuesta al gradiente en el punto:

$$\mathbf{d}^{(t)} = -\nabla f(\mathbf{x}^{(t)}) = (-2x_2 + 2x_1, -2x_1 - 2 + 4x_2)^T$$

Paso 1 (iniciación). Sea $\mathbf{x}^{(1)} = (0, 0)^T$. Por simplicidad, se comienza desde el origen, pero en la práctica puede ser conveniente hacer una adecuada selección del punto inicial.

Paso 2 (generación de la dirección de búsqueda). En la primera iteración, se tiene

$$\mathbf{x}^{(1)} = (0, 0)^T \text{ y } \mathbf{d}^{(1)} = -\nabla f(0, 0) = (0, 2)^T$$

Paso 3 (comprobación de la optimalidad). Como $\mathbf{d}^{(1)} \neq \mathbf{0}$ ésta es una dirección de descenso.

Paso 4 (búsqueda lineal). Para calcular la longitud de paso, se resuelve el problema unidimensional: Minimizar

$$Z_{LS} = f(\mathbf{x}^{(1)} + \alpha \mathbf{d}^{(1)})$$

sujeto a

$$\alpha \geq 0$$

Como

$$\mathbf{x}^{(1)} + \alpha \mathbf{d}^{(1)} = (0, 0)^T + \alpha(0, 2)^T = (0, 2\alpha)^T$$

la función objetivo de este problema es

$$g(\alpha) = f(0, 2\alpha) = -4\alpha + 8\alpha^2$$

Como $g(\alpha)$ es una función convexa, la condición suficiente de optimalidad es $g'(\alpha) = 0$. Entonces, se resuelve $g'(\alpha) = -4 + 16\alpha = 0$ y se obtiene el paso $\alpha_1 = \frac{1}{4} > 0$.

Paso 5 (actualización). Sea $\mathbf{x}^{(2)} = \mathbf{x}^{(1)} + \alpha_1 \mathbf{d}^{(1)} = (0,0)^T + \frac{1}{4}(0,2)^T = (0, \frac{1}{2})^T$.

Paso 6 (comprobación de la convergencia). Como la norma del gradiente no es suficientemente pequeña, se repite una nueva iteración con el punto $\mathbf{x}^{(2)} = (0, \frac{1}{2})^T$, y se toma $t = 1 + 1 = 2$.

Paso 2 (generación de la dirección de búsqueda). La dirección de búsqueda en el punto $\mathbf{x}^{(2)}$ es

$$\mathbf{d}^{(2)} = -\nabla f(0, \frac{1}{2}) = (1, 0)^T$$

Paso 3 (comprobación de la optimalidad). Como $\mathbf{d}^{(2)} \neq \mathbf{0}$, ésta es una dirección de descenso.

Paso 4 (búsqueda lineal). Como

$$\mathbf{x}^{(2)} + \alpha \mathbf{d}^{(2)} = (0, \frac{1}{2}) + \alpha(1, 0) = (\alpha, \frac{1}{2})^T$$

se tiene

$$g(\alpha) = f(\mathbf{x}^{(2)} + \alpha \mathbf{d}^{(2)}) = f(\alpha, \frac{1}{2}) = -\alpha + \alpha^2 - \frac{1}{2}.$$

Por tanto, el subproblema de búsqueda del paso de avance es minimizar

$$Z_{LS} = -\alpha + \alpha^2 - \frac{1}{2}$$

sujeto a

$$\alpha \geq 0$$

Como $g(\alpha) = -\alpha + \alpha^2 - \frac{1}{2}$ es una función convexa, se resuelve $g'(\alpha) = -1 + 2\alpha = 0$ y se obtiene $\alpha_2 = \frac{1}{2} > 0$.

Paso 5 (actualización). Se obtiene

$$\mathbf{x}^{(3)} = \mathbf{x}^{(2)} + \alpha_2 \mathbf{d}^{(2)} = \left(0, \frac{1}{2}\right)^T + \frac{1}{2}(1, 0)^T = \left(\frac{1}{2}, \frac{1}{2}\right)^T$$

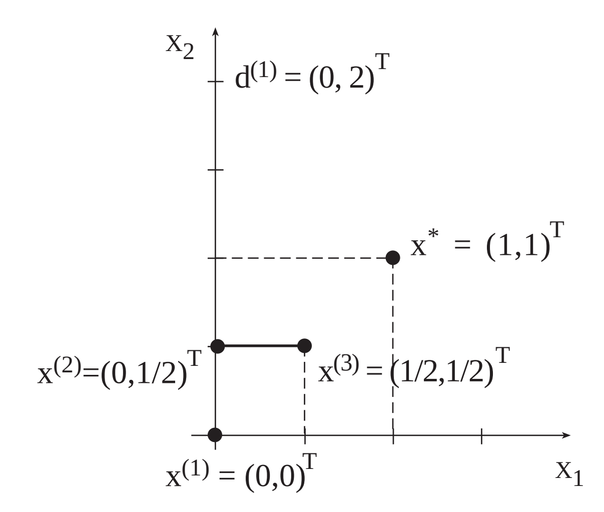


Figura 9.5: Progreso del método del gradiente.

y $t = 2 + 1 = 3$

Este procedimiento continúa hasta que se satisface el criterio de convergencia. La figura 9.5 muestra la evolución del algoritmo hacia la solución óptima.

2. Método de Newton. La dirección de búsqueda generada por este método en la iteración t es

$$\mathbf{d}^{(t)} = -\left(\nabla^2 f(\mathbf{x}^{(t)})\right)^{-1} \nabla f(\mathbf{x}^{(t)})$$

donde la matriz hessiana es

$$\nabla^2 f(x_1, x_2) = \begin{pmatrix} 2 & -2 \\ -2 & 4 \end{pmatrix}$$

cuya inversa es

$$\left(\nabla^2 f(x_1, x_2)\right)^{-1} = \begin{pmatrix} 1 & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} \end{pmatrix}.$$

Paso 1 (iniciación). Dado $\mathbf{x}^{(1)} = (0,0)^T$.

Paso 2 (generación de la dirección de búsqueda). En la primera iteración $\mathbf{x}^{(1)} = (0,0)^T$ y la dirección de búsqueda en ese punto es

$$\mathbf{d}^{(1)} = -[\nabla^2 f(\mathbf{x}^{(1)})]^{-1} \nabla f(\mathbf{x}^{(1)}) = \begin{pmatrix} -1 & -\frac{1}{2} \\ -\frac{1}{2} & -\frac{1}{2} \end{pmatrix} \begin{pmatrix} 0 \\ -2 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

Paso 3 (comprobación de optimalidad). Como $\mathbf{d}^{(1)} \neq \mathbf{0}$ y f es una función estrictamente creciente, la dirección encontrada es de descenso.

Paso 4 (búsqueda lineal). Para calcular la longitud de paso, se resuelve el problema unidimensional: Minimizar

$$Z_{LS} = f(\mathbf{x}^{(1)} + \alpha \mathbf{d}^{(1)})$$

sujeto a

$$\alpha \geq 0$$

Puesto que

$$\mathbf{x}^{(1)} + \alpha \mathbf{d}^{(1)} = (0,0)^T + \alpha(1,1)^T = (\alpha, \alpha)^T$$

se obtiene

$$f(\alpha, \alpha) = -2\alpha + \alpha^2.$$

y el valor óptimo se alcanza en $\alpha_1 = 1 > 0$.

Paso 5 (actualización). Se toma $\mathbf{x}^{(2)} = \mathbf{x}^{(1)} + \alpha_1 \mathbf{d}^{(1)} = (0,0)^T + (1,1)^T = (1,1)^T$.

Paso 6 (comprobación de la convergencia). La condición a cumplir es

$$\left| \nabla f(\mathbf{x}^{(2)}) \right| < \varepsilon$$

y se obtiene que $\nabla f(\mathbf{x}^{(2)}) = \mathbf{0}$, y por tanto se ha alcanzado la solución óptima. Esto ocurre porque el método de Newton calcula la dirección de búsqueda mediante la resolución exacta de un problema cuadrático que aproxima el problema original; por ser el problema original cuadrático, la aproximación y el original coinciden y se obtiene la solución óptima en una iteración.

3. Método quasi-Newton. En este ejemplo se empleará la fórmula DFP para actualizar las aproximaciones de la inversa de la matriz hessiana.

Paso 1 (iniciación). Se toma $\mathbf{x}^{(1)} = (0,0)^T$. En la primera iteración, el método quasi-Newton emplea como aproximación de la matriz inversa del hessiano la matriz identidad:

$$\mathbf{H}^{(1)} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

y la dirección de búsqueda es $\mathbf{d}^{(1)} = -\mathbf{H}^{(1)} \nabla f(\mathbf{x}^{(1)}) = -\nabla f(\mathbf{x}^{(1)})$. Esto significa que la primera iteración coincide con el método del gradiente. No se repetirá

este paso y se supondrá que se está en la iteración $t = 2$ y $\mathbf{x}^2 = (0, \frac{1}{2})^T$.

Paso 2 (generación de la dirección de búsqueda). Para emplear la fórmula DFP, se necesitan los siguientes vectores y matrices:

$$
\begin{aligned}
\mathbf{x}^{(1)} &= \begin{pmatrix} 0 \\ 0 \end{pmatrix} \\
\mathbf{x}^{(2)} &= \begin{pmatrix} 0 \\ \frac{1}{2} \end{pmatrix} \\
\mathbf{p}^{(1)} &= \mathbf{x}^{(2)} - \mathbf{x}^{(1)} = \begin{pmatrix} 0 \\ \frac{1}{2} \end{pmatrix} \\
\nabla f(\mathbf{x}^{(1)}) &= \begin{pmatrix} 0 \\ -2 \end{pmatrix} \\
\nabla f(\mathbf{x}^{(2)}) &= \begin{pmatrix} -1 \\ 0 \end{pmatrix} \\
\mathbf{q}^{(1)} &= \nabla f(\mathbf{x}^{(2)}) - \nabla f(\mathbf{x}^{(1)}) = \begin{pmatrix} -1 \\ 2 \end{pmatrix} \\
\mathbf{p}^{(1)}\mathbf{p}^{(1)T} &= \begin{pmatrix} 0 \\ \frac{1}{2} \end{pmatrix} (0, \frac{1}{2}) = \begin{pmatrix} 0 & 0 \\ 0 & \frac{1}{4} \end{pmatrix} \\
\mathbf{p}^{(1)T}\mathbf{q}^{(1)} &= (0, \frac{1}{2}) \begin{pmatrix} -1 \\ 2 \end{pmatrix} = 1 \\
\mathbf{H}^{(1)}\mathbf{q}^{(1)}\mathbf{q}^{(1)T}\mathbf{H}^{(1)} &= \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} -1 \\ 2 \end{pmatrix} (-1, 2) \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \\
&= \begin{pmatrix} 1 & -2 \\ -2 & 4 \end{pmatrix} \\
\mathbf{q}^{(1)T}\mathbf{H}^{(1)}\mathbf{q}^{(1)} &= (-1, 2) \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} -1 \\ 2 \end{pmatrix} = 5
\end{aligned}
$$

y se obtiene que la actualización de la aproximación a la inversa de la matriz hessiana es

$$
\begin{aligned}
\mathbf{H}^{(2)} &= \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} + \frac{1}{1} \begin{pmatrix} 0 & 0 \\ 0 & \frac{1}{4} \end{pmatrix} - \frac{1}{5} \begin{pmatrix} 1 & -2 \\ -2 & 4 \end{pmatrix} \\
&= \begin{pmatrix} \frac{4}{5} & \frac{2}{5} \\ \frac{2}{5} & \frac{9}{20} \end{pmatrix}
\end{aligned}
$$

y la dirección de búsqueda es

$$
\mathbf{d}^{(2)} = -\mathbf{H}^{(2)}\nabla f(\mathbf{x}^{(2)}) = \begin{pmatrix} -\frac{4}{5} & -\frac{2}{5} \\ -\frac{2}{5} & -\frac{9}{20} \end{pmatrix} \begin{pmatrix} -1 \\ 0 \end{pmatrix} = \begin{pmatrix} \frac{4}{5} \\ \frac{2}{5} \end{pmatrix}.
$$

Paso 3 (comprobación de optimalidad). Como $\mathbf{d}^{(2)} \neq \mathbf{0}$, es una dirección de descenso.

Paso 4 (búsqueda lineal). Para calcular la longitud de paso se resuelve el problema unidimensional consistente en minimizar

$$Z_{LS} = f(\mathbf{x}^{(2)} + \alpha \mathbf{d}^{(2)})$$

sujeto a

$$\alpha \geq 0$$

Como

$$\mathbf{x}^{(2)} + \alpha \mathbf{d}^{(2)} = \left(0, \frac{1}{2}\right)^T + \alpha \left(\frac{4}{5}, \frac{2}{5}\right)^T = \left(\frac{4\alpha}{5}, \frac{1}{2} + \frac{2\alpha}{5}\right)^T$$

se obtiene que

$$g(\alpha) = f\left(\frac{4}{5}\alpha, \frac{1}{2} + \frac{2}{5}\alpha\right) = -\frac{1}{2} - \frac{4}{5}\alpha + \frac{8}{25}\alpha^2.$$

El valor óptimo de esta búsqueda lineal es $\alpha_2 = \frac{5}{4} > 0$.

Paso 5 (actualización). Se toma

$$\mathbf{x}^{(3)} = \mathbf{x}^{(2)} + \alpha_2 \mathbf{d}^{(2)} = \left(0, \frac{1}{2}\right)^T + \frac{5}{4}\left(\frac{4}{5}, \frac{2}{5}\right)^T = (1, 1)^T.$$

Paso 6 (comprobación de la convergencia). Como $\nabla f(\mathbf{x}^{(3)}) = \mathbf{0}$, se ha alcanzado el óptimo, y el algoritmo para.

El óptimo se alcanza en dos iteraciones (dimensión de la matriz hessiana). Esto es debido al carácter cuadrático y convexo de la función objetivo.

4. Método de direcciones conjugadas. En este ejemplo los algoritmos de Fletcher–Reeves y de Polak–Ribière son equivalentes debido a que la función objetivo es cuadrática.

Paso 1 (iniciación). Se toma $\mathbf{x}^{(1)} = (0, 0)^T$. Una vez más, la primera iteración del gradiente conjugado coincide con la primera iteración del método del gradiente, ya que la primera dirección de búsqueda es $\mathbf{d}^{(1)} = -\nabla f(\mathbf{x}^{(1)})$; se supondrá pues que se está en la iteración $t = 2$ y $\mathbf{x}^2 = (0, \frac{1}{2})^T$.

Paso 2 (generación dirección de búsqueda). Se obtiene

$$\begin{array}{rcl}\nabla f(\mathbf{x}^{(1)}) & = & \begin{pmatrix} 0 \\ -2 \end{pmatrix};\ \nabla f(\mathbf{x}^{(2)}) = \begin{pmatrix} -1 \\ 0 \end{pmatrix};\ \mathbf{d}^{(n)} = -\nabla f(\mathbf{x}^{(n)}) = \begin{pmatrix} 0 \\ 2 \end{pmatrix} \\ \|\nabla f(\mathbf{x}^{(1)})\|^2 & = & 0^2 + (-2)^2 = 4 \\ \|\nabla f(\mathbf{x}^{(2)})\|^2 & = & 1^2 + 0^2 = 1 \\ \beta_1^{\mathrm{FR}} & = & \dfrac{\|\nabla f(\mathbf{x}^{(2)})\|^2}{\|\nabla f(\mathbf{x}^{(1)})\|^2} = \dfrac{1}{4}\end{array}$$

y se calcula la dirección de búsqueda como

$$\mathbf{d}^{(2)} = \nabla f(\mathbf{x}^{(2)}) + \beta_1^{\mathrm{FR}} \mathbf{d}^{(1)} = \begin{pmatrix} -1 \\ 0 \end{pmatrix} + \frac{1}{4} \begin{pmatrix} 0 \\ 2 \end{pmatrix} = \begin{pmatrix} -1 \\ \frac{1}{2} \end{pmatrix}$$

Paso 3 (comprobación de la optimalidad). Como $\mathbf{d}^{(2)} \neq \mathbf{0}$, ésta es una dirección de descenso.

Paso 4 (búsqueda lineal). Para calcular la longitud del paso, se resuelve el problema unidimensional: Minimizar

$$Z_{LS} = f(\mathbf{x}^{(2)} + \alpha \mathbf{d}^{(2)})$$

sujeto a

$$\alpha \geq 0$$

Como

$$\mathbf{x}^{(2)} + \alpha \mathbf{d}^{(2)} = (0, \frac{1}{2})^T + \alpha(1, \frac{1}{2})^T = (\alpha, \frac{1}{2} + \frac{\alpha}{2})^T$$

se obtiene

$$g(\alpha) = f(\alpha, \frac{1}{2} + \frac{\alpha}{2}) = -1/2 - \alpha + \frac{\alpha^2}{2}$$

El valor óptimo de la búsqueda lineal es pues $\alpha_2 = 1 > 0$.

Paso 5 (actualización). Se toma

$$\mathbf{x}^{(3)} = \mathbf{x}^{(2)} + \alpha_2 \mathbf{d}^{(2)} = (0, \frac{1}{2})^T + 1(1, \frac{1}{2})^T = (1, 1)^T$$

Paso 6 (comprobación de la convergencia). Puesto que se cumple $\nabla f(\mathbf{x}^{(3)}) = \mathbf{0}$, se ha alcanzado la solución óptima y el algoritmo para.

Este ejemplo ilustra el resultado teórico de que los métodos de direcciones conjugadas alcanzan la solución óptima para funciones cuadráticas convexas después de un número finito de iteraciones (igual al número de variables más 1). ∎

9.2 Algoritmos de optimización con restricciones

En esta sección se estudian problemas de optimización con restricciones. Estos problemas se pueden resolver usando fundamentalmente las siguientes familias de métodos:

1. *Métodos duales:* que resuelven el problema dual en lugar del primal.

2. *Métodos de penalizaciones:* que transforman el problema con restricciones en una sucesión de problemas sin restricciones. Las restricciones se introducen en la función objetivo mediante la llamada función de penalización–barrera y un adecuado parámetro de penalización.

3. *Método de los multiplicadores o del Lagrangiano aumentado:* éste es un método de penalizaciones (cuadráticas), en el que en lugar de incorporar las restricciones en la función objetivo se añaden a la función Lagrangiana.

4. *Métodos de direcciones factibles:* esta clase de métodos extiende los algoritmos de direcciones de descenso analizados en la sección anterior al caso de problemas con restricciones. En este contexto se fuerza a las direcciones de búsqueda, además de ser de descenso, a que den lugar a desplazamientos dentro de la región factible.

5. *Métodos de programación cuadrática secuencial:* que resuelven una sucesión de problemas cuadráticos que aproximan iterativamente al problema original.

En este libro solamente se analizan los dos primeros, el lector interesado puede consultar el resto en Bazaraa et al. [9].

9.2.1 Métodos duales

En esta sección se estudia la resolución del PPNL mediante la resolución de su problema dual. La principal motivación es que el problema dual, en la mayoría de los casos, requiere la maximización de una función cóncava sobre un conjunto convexo simple, y entonces, no posee máximos locales diferentes a los globales. Las dos cuestiones fundamentales que se deben responder para validar este esquema son:

1. ¿Cómo obtener la solución del problema primal a partir de la del dual?

2. ¿Cómo resolver el problema dual?

Considérese el problema: Minimizar

$$Z_P = f(\mathbf{x})$$

sujeto a

$$\begin{array}{rcl}\mathbf{h}(\mathbf{x}) & = & \mathbf{0}\\ \mathbf{g}(\mathbf{x}) & \leq & \mathbf{0}.\end{array}$$

donde $f : \mathbb{R}^n \to \mathbb{R}$, $\mathbf{h} : \mathbb{R}^\ell \to \mathbb{R}$ y $\mathbf{g} : \mathbb{R}^n \to \mathbb{R}^m$ son continuas. Su problema dual es: Maximizar

$$Z_D = \theta(\boldsymbol{\lambda}, \boldsymbol{\mu})$$

sujeto a

$$\boldsymbol{\mu} \geq \mathbf{0}$$

donde la función dual se define por

$$\theta(\boldsymbol{\lambda}, \boldsymbol{\mu}) \;=\; \text{Infimo}_{\mathbf{x}} \left(f(\mathbf{x}) + \sum_{k=1}^{\ell} \lambda_k h_k(\mathbf{x}) + \sum_{j=1}^{m} \mu_j g_j(\mathbf{x}) \right)$$

cuyo dominio de definición

$$D = \{(\boldsymbol{\lambda}, \boldsymbol{\mu}) | \boldsymbol{\mu} \geq \mathbf{0}, \text{ existe el mínimo } \mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu})\},$$

esto es, D es el conjunto de vectores $(\boldsymbol{\lambda}, \boldsymbol{\mu})$ tales que $\boldsymbol{\mu}$ es no negativo y para los cuales $\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu})$ tiene ínfimo finito. Por simplicidad, se supondrá que

$$D = \mathbb{R}^\ell \times (\mathbb{R}^m)^+ = \{(\boldsymbol{\lambda}, \boldsymbol{\mu}) | \boldsymbol{\mu} \geq \mathbf{0}\}$$

Para evaluar la función dual para $\boldsymbol{\lambda}^{(t)}$ y $\boldsymbol{\mu}^{(t)}$, es necesario resolver el problema

$$\text{minimizar}_{\mathbf{x}} \;\; \mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}^{(t)}, \boldsymbol{\mu}^{(t)})$$

que se denomina *problema Lagrangiano.*

Para discutir la primera cuestión (¿cómo obtener la solución del primal a partir del dual?), se define el conjunto

$$X(\boldsymbol{\lambda}, \boldsymbol{\mu}) = \{\mathbf{x} | \mathbf{x} \text{ minimiza } \mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu})\}$$

Si no existe la llamada holgura dual, esto es, si $Z_D^* = Z_P^*$, y se es capaz de resolver el problema dual, y se obtiene como valores óptimos $(\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*)$, entonces cualquier punto factible $\mathbf{x}^* \in X(\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*)$ resuelve el problema primal. El teorema 9.1 justifica que para problemas convexos no exista holgura dual, y por tanto, se puede obtener la solución del problema primal conociendo la del dual. El teorema siguiente, basándose en la diferenciabilidad de la función dual, asegura que no existe holgura dual en problemas convexos, y por tanto permite responder a la primera cuestión.

Teorema 9.1 (condición suficiente para obtener una solución del primal dada una del dual). *Dada $(\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*)$, una solución del problema dual, y suponiendo que $\theta(\boldsymbol{\lambda}, \boldsymbol{\mu})$ es diferenciable en $(\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*)$, entonces cualquier elemento $\mathbf{x}^* \in X(\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*)$ resuelve el problema primal.* ■

Teorema 9.2 (diferenciabilidad de la función dual). *La función objetivo del problema dual es diferenciable en un punto* $(\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*)$ *si y sólo si* h_k *y* g_j *son constantes en* $X(\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*)$. *En este caso, las derivadas parciales de* θ *son*

$$\left.\frac{\partial\theta(\boldsymbol{\lambda}, \boldsymbol{\mu})}{\partial\lambda_k}\right|_{(\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*)} = h_k(\mathbf{x}), \text{ para cualquier } \mathbf{x} \in X(\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*)$$
$$\left.\frac{\partial\theta(\boldsymbol{\lambda}, \boldsymbol{\mu})}{\partial\mu_j}\right|_{(\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*)} = g_j(\mathbf{x}), \text{ para cualquier } \mathbf{x} \in X(\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*).$$

■

Una condición que garantiza el cumplimiento del teorema anterior, esto es, que las restricciones sean constantes en el conjunto $X(\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*)$, es que el conjunto $X(\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*)$ contenga un único elemento, esto es que $\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*)$ tenga un único mínimo $\mathbf{x}(\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*)$. Para problemas convexos, ésto se verifica cuando la función objetivo es estrictamente convexa y en dicho caso la función lagrangiana $\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu})$ también es estrictamente convexa para todo $(\boldsymbol{\lambda}, \boldsymbol{\mu})$. Entonces, el problema Lagrangiano tiene una única solución (si ésta existe) (véase el Teorema 8.1).

De acuerdo con los resultados anteriores, bajo la suposición de que $X(\boldsymbol{\lambda}, \boldsymbol{\mu})$ es unitario, el problema dual es no restringido, excepto por la condición $\boldsymbol{\mu} \geq \mathbf{0}$, y $\theta(\boldsymbol{\lambda}, \boldsymbol{\mu})$ diferenciable en D. Entonces, un método del tipo gradiente es apropiado para maximizar la función dual. Esto resuelve la segunda cuestión: ¿cómo resolver el problema dual?

De las restricciones $\boldsymbol{\mu} \geq \mathbf{0}$ se tiene que la dirección de ascenso

$$\mathbf{d}^{(t)} = (\mathbf{d}_{\boldsymbol{\lambda}}^{(t)}, \mathbf{d}_{\boldsymbol{\mu}}^{(t)}) = (\nabla_{\boldsymbol{\lambda}}\theta(\boldsymbol{\lambda}^{(t)}, \boldsymbol{\mu}^{(t)}), \nabla_{\boldsymbol{\mu}}\theta(\boldsymbol{\lambda}^{(t)}, \boldsymbol{\mu}^{(t)}))$$

podría ser infactible (si uno se moviese fuera de la región factible), por ese motivo se emplea como dirección de búsqueda la proyección del gradiente en la región factible $\{(\boldsymbol{\lambda}, \boldsymbol{\mu}) \in \mathbb{R}^{\ell} \times \mathbb{R}^{m} | \boldsymbol{\mu} \geq \mathbf{0}\}$, obteniéndose un ascenso y una factibilidad a lo largo de la dirección de búsqueda. Este método se denomina *método del gradiente proyectado de Rosen para problemas con restricciones lineales* (véase Rosen [96]) particularizado para las restricciones $\boldsymbol{\mu} \geq \mathbf{0}$.

El algoritmo del gradiente, modificado para considerar las restricciones $\boldsymbol{\mu} \geq \mathbf{0}$, aplicado a la maximización $\theta(\boldsymbol{\lambda}, \boldsymbol{\mu})$, se describe a continuación.

Algoritmo 9.3 (algoritmo modificado del gradiente).

Paso 1 (iniciación). Se elige un punto inicial $(\boldsymbol{\lambda}^{(1)}, \boldsymbol{\mu}^{(1)}) \in D$, y se toma $t = 1$.

Paso 2 (problema lagrangiano). Resolver el problema lagrangiano con $\boldsymbol{\lambda} = \boldsymbol{\lambda}^{(t)}$ y $\boldsymbol{\mu} = \boldsymbol{\mu}^{(t)}$, obteniendo $\mathbf{x}(\boldsymbol{\lambda}^{(t)}, \boldsymbol{\mu}^{(t)})$, y

$$\theta(\boldsymbol{\lambda}^{(t)}, \boldsymbol{\mu}^{(t)}) = \mathcal{L}(\mathbf{x}(\boldsymbol{\lambda}^{(t)}, \boldsymbol{\mu}^{(t)}), \boldsymbol{\lambda}^{(t)}, \boldsymbol{\mu}^{(t)})$$

Paso 3 (evaluación del gradiente). Evaluar el gradiente (véase el capítulo 8)

$$\begin{aligned} \nabla_{\boldsymbol{\mu}}\theta(\boldsymbol{\lambda}^{(t)},\boldsymbol{\mu}^{(t)}) &= \mathbf{g}(\mathbf{x}(\boldsymbol{\lambda}^{(t)},\boldsymbol{\mu}^{(t)})) \\ \nabla_{\boldsymbol{\lambda}}\theta(\boldsymbol{\lambda}^{(t)},\boldsymbol{\mu}^{(t)}) &= \mathbf{h}(\mathbf{x}(\boldsymbol{\lambda}^{(t)},\boldsymbol{\mu}^{(t)})) \end{aligned}$$

Paso 4 (comprobación de la optimalidad). Si

$$\nabla_{\boldsymbol{\mu}}\theta(\boldsymbol{\lambda}^{(t)},\boldsymbol{\mu}^{(t)}) = \mathbf{0} \text{ y } \nabla_{\boldsymbol{\lambda}}\theta(\boldsymbol{\lambda}^{(t)},\boldsymbol{\mu}^{(t)}) = \mathbf{0}$$

entonces *parar*. En caso contrario, continuar.

Paso 5 (generación de la dirección de búsqueda). La dirección de búsqueda está dada por $\mathbf{d}^{(t)} = (\mathbf{d}_{\boldsymbol{\lambda}}^{(t)}, \mathbf{d}_{\boldsymbol{\mu}}^{(t)})$, con

$$\begin{aligned} \mathbf{d}_{\mu_j}^{(t)} &= \begin{cases} \left.\dfrac{\partial\theta(\boldsymbol{\lambda},\boldsymbol{\mu})}{\partial\mu_j}\right|_{(\boldsymbol{\lambda}^{(t)},\boldsymbol{\mu}^{(t)})} & \text{si } \mu_j^{(t)} > 0 \\ \max\left\{0, \left.\frac{\partial\theta(\boldsymbol{\lambda},\boldsymbol{\mu})}{\partial\mu_j}\right|_{\left(\boldsymbol{\lambda}^{(t)},\boldsymbol{\mu}^{(t)}\right)}\right\} & \text{si } \mu_j^{(t)} = 0 \end{cases} \\ \mathbf{d}_{\boldsymbol{\lambda}}^{(t)} &= \nabla_{\boldsymbol{\lambda}}\theta(\boldsymbol{\lambda}^{(t)},\boldsymbol{\mu}^{(t)}) \end{aligned}$$

Paso 6 (búsqueda lineal). Encontrar una longitud de paso, α_t, que resuelva el problema unidimensional: Maximizar con respecto a α

$$\theta\left((\boldsymbol{\lambda}^{(t)},\boldsymbol{\mu}^{(t)}) + \alpha\mathbf{d}^{(t)}\right)$$

sujeto a

$$\begin{aligned} (\boldsymbol{\lambda}^{(t)},\boldsymbol{\mu}^{(t)}) + \alpha\mathbf{d}^{(t)} &\in D \\ \alpha &\geq 0 \end{aligned}$$

Paso 7 (actualizar). Se toma $(\boldsymbol{\lambda}^{(t+1)},\boldsymbol{\mu}^{(t+1)}) = (\boldsymbol{\lambda}^{(t)},\boldsymbol{\mu}^{(t)}) + \alpha_t\mathbf{d}^{(t)}$, y $t = t+1$.

Paso 8 (comprobación de la convergencia). Si se cumple la condición

$$||\boldsymbol{\lambda}^{(t+1)} - \boldsymbol{\lambda}^{(t)}|| \leq \epsilon \text{ y } ||\boldsymbol{\mu}^{(t+1)}) - \boldsymbol{\mu}^{(t)})|| \leq \epsilon$$

entonces *parar* el proceso. En caso contrario ir al paso 2.

∎

Nótese que la evaluación de la función dual requiere resolver el problema lagrangiano, lo que origina que las búsquedas lineales exactas requieran un tiempo de cálculo no razonable. Esto hace necesario elegir el paso α_t de forma sencilla

y de modo que se produzca un incremento de la función dual, por tanto, se selecciona el paso de modo que

$$\theta\left((\boldsymbol{\lambda}^{(t)}, \boldsymbol{\mu}^{(t)}) + \alpha_t \mathbf{d}^{(t)}\right) > \theta(\boldsymbol{\lambda}^{(t)}, \boldsymbol{\mu}^{(t)}) \tag{9.18}$$

y el coste computacional sea bajo.

Si θ es diferenciable, y si $(\boldsymbol{\lambda}^{(t)}, \boldsymbol{\mu}^{(t)})$ no maximiza θ, existe un α_t que cumple (9.18). En algunos casos, las búsquedas lineales llegan a ser simples debido a la estructura especial de la función dual (véase el ejemplo 9.4).

Es importante destacar que el método del gradiente proyectado de Rosen converge a una solución global cuando se maximizan funciones cóncavas diferenciables con restricciones de la forma $\boldsymbol{\mu} \geq 0$ (véase [33]). No obstante, en muchos casos prácticos la función dual no es diferenciable.

Ejemplo 9.4 (método dual). Considérese el problema: Minimizar

$$Z = \sum_{i=1}^{m}\sum_{j=1}^{n} f_{ij}(T_{ij}) = \sum_{i=1}^{m}\sum_{j=1}^{n} T_{ij}(\frac{\log T_{ij}}{t_{ij}} - 1) \tag{9.19}$$

sujeto a

$$\sum_{j=1}^{n} T_{ij} = r_i, \quad i = 1, \ldots, m \tag{9.20}$$

$$\sum_{i=1}^{m} T_{ij} = c_j, \quad j = 1, \ldots, n \tag{9.21}$$

$$T_{ij} \geq 0, \quad i = 1, \ldots, m, \; j = 1, \ldots, n \tag{9.22}$$

donde t_{ij}, r_i, y c_j son constantes, y T_{ij} son las variables del problema. El problema (9.19)–(9.22) es el problema primal. A continuación se formula el problema dual. Considérese $T = \{\mathbf{T}|T_{ij} \geq 0\}$, e introdúzcanse los multiplicadores $\lambda_i^r; i = 1, \ldots, m$ para las restricciones (9.20) y $\lambda_j^c; j = 1, \ldots, n$ para las restricciones (9.21).

La función Lagrangiana es

$$\begin{aligned}
\mathcal{L}(\mathbf{T}, \boldsymbol{\lambda}^r, \boldsymbol{\lambda}^c) &= \sum_{i=1}^{m}\sum_{j=1}^{n} f_{ij}(T_{ij}) + \sum_{i=1}^{m} \lambda_i^r \left(\sum_{j=1}^{n} T_{ij} - r_i\right) \\
&\quad + \sum_{j=1}^{n} \lambda_j^c \left(\sum_{i=1}^{m} T_{ij} - c_j\right) \\
&= \sum_{i=1}^{m}\sum_{j=1}^{n} \left(f(T_{ij}) + (\lambda_i^r + \lambda_j^c) T_{ij}\right) \\
&\quad - \left(\sum_{i=1}^{m} r_i \lambda_i^r + \sum_{j=1}^{n} c_j \lambda_j^c\right).
\end{aligned} \tag{9.23}$$

Nótese que el segundo término no depende de las variables T_{ij}.

Para calcular la función dual se debe resolver el problema: Minimizar con respecto a $\mathbf{T}$

$$Z = \mathcal{L}(\mathbf{T}, \boldsymbol{\lambda}^r, \boldsymbol{\lambda}^c) \tag{9.24}$$

sujeto a

$$\mathbf{T} \in T \tag{9.25}$$

Entonces, la función dual es

$$\theta(\boldsymbol{\lambda}^r, \boldsymbol{\lambda}^c) = \min_{\mathbf{T} \in T} \mathcal{L}(\mathbf{T}, \boldsymbol{\lambda}^r, \boldsymbol{\lambda}^c) \tag{9.26}$$

Este problema es separable en las variables T_{ij}, y por tanto la solución de (9.24)–(9.25) puede ser obtenida resolviendo los siguientes $n \times m$ problemas unidimensionales: Minimizar

$$Z_{ij} = f_{ij}(T_{ij}) + \lambda_{ij} T_{ij}$$

sujeto a

$$T_{ij} \geq 0$$

donde $\lambda_{ij} = \lambda_i^r + \lambda_j^c$.

Este es un problema convexo debido a que la función objetivo es la suma de funciones convexas y las restricciones son lineales. La condición suficiente de optimalidad para una solución factible T_{ij} es que el gradiente sea cero. Esta condición conduce a la ecuación

$$f'_{ij}(T^*_{ij}) = -\lambda_{ij}$$

cuya solución es

$$T^*_{ij} \quad = \quad t_{ij} \exp(-\lambda_{ij}) > 0 \tag{9.27}$$

y entonces, el valor óptimo de la función objetivo es

$$\begin{array}{rcl} Z^*_{ij} & = & f_{ij}(T^*_{ij}) + \lambda_{ij} T^*_{ij} \\ & = & T^*_{ij} \left(\log(t_{ij} \exp(-\lambda_{ij})/t_{ij}) - 1\right) + \lambda_{ij} T^*_{ij} \\ & = & -t_{ij} \exp(-\lambda_{ij}) \end{array} \tag{9.28}$$

Sustituyendo T^*_{ij} en la función Lagrangiana, se obtiene

$$\begin{array}{rcl} \theta(\boldsymbol{\lambda}^r, \boldsymbol{\lambda}^c) & = & \mathcal{L}(T^*_{ij}, \lambda_i^r, \lambda_j^c) = \displaystyle\sum_{i=1}^{m} \sum_{j=1}^{n} Z^*_{ij} - \left(\sum_{i=1}^{m} r_i \lambda_i^r + \sum_{j=1}^{n} c_j \lambda_j^c \right) \\ & = & -\displaystyle\sum_{i=1}^{m} \sum_{j=1}^{n} t_{ij} \exp(-\lambda_{ij}) - \sum_{i=1}^{m} r_i \lambda_i^r - \sum_{j=1}^{n} c_j \lambda_j^c \end{array}$$

y el problema dual puede formularse como sigue: Maximizar

$$\theta(\boldsymbol{\lambda}^r, \boldsymbol{\lambda}^c)$$

Tabla 9.3: Datos para el problema del ejemplo 9.4.

i o j	(r_i)	(c_j)
1	12000	6750
2	10500	7300
3	3800	10000
4	7700	9950

	t_{ij}			
	1	2	3	4
1	–	60	275	571
2	50	–	410	443
3	123	61	–	47
4	205	265	75	–

Tabla 9.4: Resultados para el problema dual del ejemplo 9.4

(t)	$(\boldsymbol{\lambda}^r)^{(t)}$				$\nabla_{\boldsymbol{\lambda}^r}\theta$			
1	0	0	0	0	-11094.00	-9597.00	-3569.00	-7155.00
2	-1.54	-1.33	-0.50	-0.99	2507.84	1329.54	-2784.66	-3760.61
3	-1.31	-1.21	-0.76	-1.34	-3338.84	-2237.14	-2113.47	158.99
100	-1.68	- 1.52	-1.15	-0.92	0.00	0.00	0.00	0.00
(t)	$(\boldsymbol{\lambda}^c)^{(t)}$				$\nabla_{\boldsymbol{\lambda}^c}\theta$			
1	0	0	0	0	-6372.00	-6914.00	-9240.00	-8889.00
2	-0.88	-0.96	-1.28	-1.23	-4461.50	-4439.23	960.95	5231.90
3	-1.30	-1.37	-1.19	-0.75	-2282.42	-1891.17	-1181.18	-2175.70
100	-1.78	-1.82	-1.04	-0.64	0.00	0.00	0.00	0.00

Tabla 9.5: Resultados adicionales del problema dual del ejemplo 9.4

(t)	$\|\nabla\theta\|_t$	α_t	$\theta((\boldsymbol{\lambda}^r, \boldsymbol{\lambda}^c)^{(t+1)})$
1	23063.15	1.3e-004	-2585.00
2	9892.49	9.3e-005	48793.09
3	5963.18	7.3e-005	53273.57
100	0.00	4.8e-005	55709.91

En lo que sigue se aplica este método para resolver el problema (9.19)–(9.22) empleando los datos mostrados en la tabla 9.3.

Un resumen de los cálculos y la solución se muestra en la tabla 9.4, que proporciona los valores de las variables duales, y el gradiente de la función dual. La tabla 9.5 contiene la función dual, la norma del gradiente de la función dual y la longitud de paso usada en cada iteración. Finalmente, la tabla 9.6 proporciona la solución del problema (valores de las variables primales). Nótese que el gradiente de la función dual se aproxima a cero, y la función dual va incrementando su valor hasta alcanzar el óptimo (el gradiente llega a valer cero).

■

Tabla 9.6: Solución primal del ejemplo 9.4

	T_{ij}			
	1	2	3	4
1	–	1997.89	4176.46	5825.64
2	1364.38	–	5293.38	3842.24
3	2322.86	1195.02	–	282.12
4	3062.76	4107.08	530.16	–

9.2.2 Métodos de penalizaciones

Los métodos de penalizaciones transforman el problema original con restricciones en una sucesión de problemas sin restricciones mediante las llamadas funciones de penalización. La idea de convertir problemas con restricciones en problemas sin restricciones es muy adecuada, debido en parte a que estos problemas se pueden resolver muy eficazmente. Todos los métodos de penalizaciones tratan las restricciones de igualdad de la misma manera, diferenciándose en el modo de considerar las restricciones de desigualdad, dando lugar a dos tipos de métodos:

1. *Métodos de puntos exteriores.* La sucesión de soluciones de los problemas sin restricciones son puntos infactibles para el problema original.

2. *Métodos de puntos interiores o de barrera.* En esta clase de métodos la sucesión de soluciones generadas son puntos factibles. Un caso particular de esta clase de métodos es el *método del punto interior* para resolver problemas de programación lineal, que será analizado en este capítulo.

Métodos de penalizaciones exteriores

En el *método de penalización exterior*, la penalización impone que la función objetivo incremente su valor cuando el valor de $\mathbf{x}$ se aleja de la región factible. Se actualizan los parámetros de penalización haciendo que la sucesión de soluciones de los problemas de optimización sin restricciones asociados converja a la solución óptima. Está sucesión está contenida en el exterior de la región factible.

Considérese el problema de optimización: Minimizar

$$Z = f(\mathbf{x})$$

sujeto a

$$\begin{array}{rcl} \mathbf{h}(\mathbf{x}) & = & \mathbf{0} \\ \mathbf{g}(\mathbf{x}) & \leq & \mathbf{0} \end{array} \tag{9.29}$$

donde $f : \mathbb{R}^n \to \mathbb{R}$, $\mathbf{g} : \mathbb{R}^n \to \mathbb{R}^m$ y $\mathbf{h} : \mathbb{R}^\ell \to \mathbb{R}$ son continuas. Se denota por S la región factible.

La *función de penalización exterior* es

$$P(\mathbf{x}; r) = f(\mathbf{x}) + r\psi(\mathbf{h}(\mathbf{x}), \mathbf{g}(\mathbf{x}))$$

donde r es el *parámetro de penalización* y $\psi : \mathbb{R}^{\ell+m} \to \mathbb{R}$ es la *función de penalización* (una función de las restricciones), que cumple

$$\begin{aligned} \psi(\mathbf{h}(\mathbf{x}), \mathbf{g}(\mathbf{x})) = 0; \ \ \forall \mathbf{x} \in S \\ \psi(\mathbf{h}(\mathbf{x}), \mathbf{g}(\mathbf{x})) > 0; \ \ \forall \mathbf{x} \notin S \end{aligned}$$

Teorema 9.3 (convergencia del método de penalización exterior). *Sea* $\{r_t\}$ *una sucesión de números positivos divergiendo a* $+\infty$ *y* $\psi : \mathbb{R}^{\ell+m} \to \mathbb{R}$ *una función de penalización exterior continua. Se define la sucesión*

$$\mathbf{x}^{(t)} = \textit{arg minimiza } \{f(\mathbf{x}) + r_t \psi(\mathbf{h}(\mathbf{x}), \mathbf{g}(\mathbf{x}))\}, \ t = 1, 2, \ldots \tag{9.30}$$

y se supone que para cada r_t*, existe una solución óptima del problema asociado.*

Cada punto de acumulación (punto límite) de la sucesión $\{\mathbf{x}^{(t)}\}$ *es un mínimo global del problema original con restricciones. En particular, si el problema original tiene un único óptimo, entonces la sucesión* $\{\mathbf{x}^{(t)}\}$ *converge a dicho óptimo.*

■

Una demostración de este teorema se puede encontrar en Luenberger [68].

La función de penalización más común es

$$P_{\alpha,\beta}(\mathbf{x}; r) = f(\mathbf{x}) + r \left[\sum_{k=1}^{\ell} |h_k(\mathbf{x})|^{\alpha} + \sum_{j=1}^{m} |g_j(\mathbf{x})|_+^{\beta} \right]; \ \alpha, \beta \geq 1$$

donde $|x|_+ = \max\{x, 0\}$ es el *operador de no negatividad.* Dos casos particulares son

1. Función de penalización valor absoluto

$$P_1(\mathbf{x}; r) = f(\mathbf{x}) + r \left[\sum_{k=1}^{\ell} |h_k(\mathbf{x})| + \sum_{j=1}^{m} |g_j(\mathbf{x})|_+ \right]$$

2. Función de penalización cuadrática

$$P_2(\mathbf{x}; r) = f(\mathbf{x}) + r \left[\sum_{k=1}^{\ell} |h_k(\mathbf{x})|^2 + \sum_{j=1}^{m} |g_j(\mathbf{x})|_+^2 \right]$$

Los algoritmos de penalización exterior tienen la siguiente estructura.

Algoritmo 9.4 (método de penalización exterior).

Paso 1 (iniciación). Se elige como punto inicial $\mathbf{x}^{(0)}$ y como parámetro de penalización $r_1 > 0$ y se hace $t = 1$. Sea $\varepsilon > 0$ el parámetro de tolerancia y $\eta > 1$ un número fijo.

Paso 2 (subproblema). Se resuelve el problema: Minimizar

$$f(\mathbf{x}) + r_t\psi(\mathbf{h}(\mathbf{x}), \mathbf{g}(\mathbf{x})) \tag{9.31}$$

mediante un adecuado método de descenso, usando como punto inicial $\mathbf{x}^{(t-1)}$. La solución del problema (9.31) se denomina $\mathbf{x}^{(t)}$.

Paso 3 (criterio de parada). Si $\|\mathbf{x}^{(t)} - \mathbf{x}^{(t-1)}\| < \varepsilon$, entonces se *para* el proceso. En caso contrario, se va al paso 4.

Paso 4 (actualizar contadores). Se hace $r_{t+1} = \eta r_t$, $t = t+1$ y se va al paso 2.

∎

Los métodos de penalizaciones tienen las siguientes propiedades:

1. $P(\mathbf{x}(r), r)$ y $f(\mathbf{x}(r))$, donde $\mathbf{x}(r)$ minimiza $P(\mathbf{x}(r), r)$, son funciones no decrecientes en r.
2. $\psi\left(\mathbf{h}(\mathbf{x}(r)), \mathbf{g}(\mathbf{x}(r))\right)$ es una función no decreciente en la variable r.
3. Se cumple la condición:

$$\lim_{t\to+\infty} r_t\psi\left(\mathbf{h}(\mathbf{x}^{(t)}), \mathbf{g}(\mathbf{x}^{(t)})\right) = 0$$

Estas propiedades son analizadas en detalle en Luenberger [68].

El siguiente ejemplo ilustra el método de penalización interior.

Ejemplo 9.5 (método de penalización exterior). Considérese el problema: Minimizar

$$Z = (x_1 - 6)^2 + (x_2 - \frac{17}{2})^2$$

sujeto a

$$x_1^2 - x_2 = 0$$

Nótese que en la iteración t, el problema a resolver para obtener el valor de $x^{(t)}$ es: Minimizar

$$Z = (x_1 - 6)^2 + (x_2 - 17/2)^2 + r_t|x_1^2 - x_2|^\beta$$

La tabla 9.7 resume los cálculos efectuados empleando el método de penalización para dos funciones de penalización, definidas por los parámetros $\beta = 2$ y $\beta = 1$. El punto inicial es $\mathbf{x}^{(0)} = (0, 0)$ y los valores de los parámetros $r_1 = 0.001$ y $\eta = 10$. La figura 9.6 ilustra la evolución del algoritmo para estos dos métodos.

∎

Tabla 9.7: Resultados del ejemplo 9.5

β	t	r_t	$x_1^{(t)}$	$x_2^{(t)}$	$f(x^{(t)})$	$P(\mathbf{x}^{(t)}, r_t)$	$r_t\psi(h(\mathbf{x}^{(t)}))$
2	1	0.001	5.723	8.524	0.0775	0.664	5.869×10^{-1}
	2	0.010	4.716	8.636	1.666	3.518	1.852×10^{-0}
	3	0.100	3.518	8.853	6.283	7.526	1.244×10^{-0}
	4	1.000	3.074	8.976	8.786	9.012	2.264×10^{-1}
	5	10.00	3.008	8.997	9.200	9.225	2.474×10^{-2}
	6	100.0	3.001	8.000	9.245	9.248	2.498×10^{-3}
	7	1,000	3.000	9.000	9.250	9.250	2.500×10^{-4}
	8	10,000	3.000	9.000	9.250	9.250	2.502×10^{-5}
1	1	0.001	5.994	8.501	0.000	0.0275	2.7428×10^{-2}
	2	0.010	5.941	8.505	0.000	0.2714	2.679×10^{-1}
	3	0.100	5.455	8.550	0.300	2.420	2.120×10^{-0}
	4	1.000	3.000	9.000	9.250	9.250	1.305×10^{-5}
	5	10.00	3.000	9.000	9.250	9.250	6.202×10^{-9}
	6	100.0	3.000	9.000	9.250	9.250	2.132×10^{-12}
	7	1,000	3.000	9.000	9.250	9.250	0
	8	10,000	3.000	9.000	9.250	9.250	0
Exacto	∞		3.000	9.000	9.250	9.250	0

Métodos de penalización interior o métodos barrera

Considérese el problema de optimización con restricciones de desigualdad: Minimizar

$$Z = f(\mathbf{x})$$

sujeto a

$$\mathbf{g}(\mathbf{x}) \leq \mathbf{0}$$

donde $f : \mathbb{R}^n \to \mathbb{R}$, $\mathbf{g} : \mathbb{R}^n \to \mathbb{R}^m$ son funciones continuas, y $S = \{\mathbf{x}|\mathbf{g}(\mathbf{x}) \leq \mathbf{0}\}$ es la región factible. Se supone que $S^- = \{\mathbf{x}|\mathbf{g}(\mathbf{x}) < \mathbf{0}\}$ es no vacío, y que la clausura de S^- es el conjunto S.

Considérese la función objetivo

$$P(\mathbf{x}; r) = f(\mathbf{x}) + r\phi\left(\mathbf{g}(\mathbf{x})\right)$$

donde r es el *parámetro de penalización* y ϕ es la *función de penalización interior*, esto es, una función de las restricciones. Esta función forma una barrera que tiende a infinito a lo largo de la frontera de la región factible, lo que tiende a favorecer la selección de puntos factibles sobre los infactibles cuando se emplean métodos de optimización sin restricciones inicializados en puntos factibles.

Las funciones de penalización interior son funciones definidas en $(\mathbb{R}^m)^- = \{\mathbf{y} \in \mathbb{R}^m|\mathbf{y} < \mathbf{0}\}$, que cumplen las siguientes condiciones:

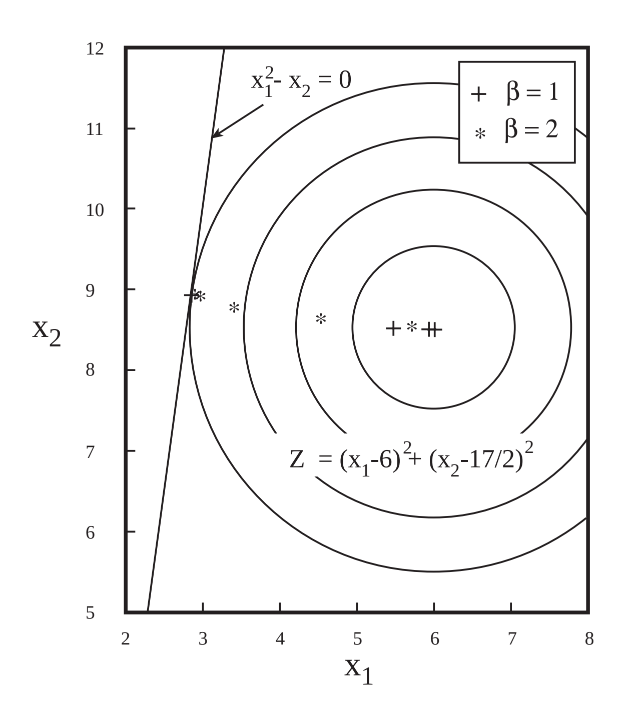


Figura 9.6: Ilustración del progreso de los métodos de penalización exterior: $\beta = 1$ y $\beta = 2$.

1. Son positivas en el interior de la región factible

$$\phi\left(\mathbf{g}(\mathbf{x})\right) > 0; \quad \forall \mathbf{x} \in S^{-} \tag{9.32}$$

2. Tienden a ∞ en la frontera de la región factible

$$\lim_{\mathbf{x} \to S - S^{-}} \phi(\mathbf{g}(\mathbf{x})) = +\infty \tag{9.33}$$

Algunos ejemplos de funciones barrera, definidas para las restricciones $\mathbf{g}(\mathbf{x}) \leq \mathbf{0}$, son

1. La *función de penalización barrera logarítmica*

$$\phi(\mathbf{g}(\mathbf{x})) = -\sum_{j=1}^{m} \log(-g_j(\mathbf{x}))$$

2. La *función de penalización barrera inversa*

$$\phi(\mathbf{g}(\mathbf{x})) = \sum_{j=1}^{m} \frac{-1}{g_j(\mathbf{x})}$$

Teorema 9.4 (convergencia de los métodos barrera). *Sea* $\{r_t\}$ *una sucesión de números positivos convergiendo a cero, y sea* $\phi : \mathbb{R}^m \to \mathbb{R}$ *una función de penalización interior, se define*

$$\mathbf{x}^{(t)} = \text{arg minimiza}_{\mathbf{x}\in S^-} \{f(\mathbf{x}) + r_t\phi(g(\mathbf{x}))\}, \; t = 1, 2, \ldots \tag{9.34}$$

Cada punto de acumulación de la sucesión generada por el método barrera, $\{\mathbf{x}^{(t)}\}$*, es un mínimo global del problema. En particular, si el problema original tiene un único óptimo, entonces la sucesión* $\{\mathbf{x}^{(t)}\}$ *converge a él.* ■

Una demostración de este teorema se encuentra en Luenberger [68].

Un algoritmo barrera tiene la siguiente estructura.

Algoritmo 9.5 (método barrera).

Paso 1 (iniciación). Se elige un punto inicial $\mathbf{x}^{(0)}$ que cumpla $\mathbf{x}^{(0)} \in S^-$, un parámetro de penalización inicial $r_1 > 0$ y se hace $t = 1$. Sea $\varepsilon > 0$ un número suficientemente pequeño que representa la tolerancia, y $\eta \in (0, 1)$ un número fijo.

Paso 2 (subproblema). Se resuelve, mediante un adecuado método de descenso, donde $\mathbf{x}^{(t-1)}$ es el punto inicial, el problema: Minimizar

$$Z = \{f(\mathbf{x}) + r_t\phi(\mathbf{g}(\mathbf{x}))\}$$

sujeto a

$$\mathbf{x} \in S^-$$

cuya solución óptima es $\mathbf{x}^{(t)}$.

Paso 3 (criterio de parada). Si $\|\mathbf{x}^{(t)} - \mathbf{x}^{(t-1)}\| < \varepsilon$, se para el proceso (existen criterios de parada alternativos a éste). En caso contrario, se va al paso 4.

Paso 4 (actualización de contadores). Se hace $r_{t+1} = \eta r_t$, $t = t + 1$, y se va al paso 2. ■

El ejemplo siguiente ilustra el método de penalización interior o barrera.

Ejemplo 9.6 (método barrera). Considérese el problema: Minimizar

$$Z = -5x_1 + x_1^2 - 8x_2 + 2x_2^2$$

sujeto a

$$3x_1 + 2x_2 \leq 6$$

La función objetivo más una penalización de barrera logarítmica es

$$P(\mathbf{x}, r) = -5x_1 + x_1^2 - 8x_2 + 2x_2^2 - r\log(6 - 3x_1 + 2x_2)$$

Los puntos estacionarios de $P(\mathbf{x}, r)$ en función de r son

$$
\begin{aligned}
\frac{\partial P}{\partial x_1} &= 2x_1 - 5 + \frac{3r}{6 - 3x_1 - 2x_2} = 0 \\
\frac{\partial P}{\partial x_2} &= 4x_2 - 8 + \frac{2r}{6 - 3x_1 - 2x_2} = 0
\end{aligned}
$$

que conduce a $2x_1 - 6x_2 + 7 = 0$. Por consiguiente

$$
\begin{aligned}
2x_1 - 5 - \frac{3r}{6 - 3x_1 - 2(-7/6 - x_1/3)} = 0 \\
4x_2 - 8 - \frac{2r}{6 - 3(7/2 + 3x_2) - 2x_2} = 0
\end{aligned}
$$

que es equivalente a

$$
\begin{aligned}
22x_1^2 - 77x_1 - 9r + 55 &= 0 \\
22x_2^2 - 77x_2 - r + 66 &= 0
\end{aligned}
\tag{9.35}
$$

La solución de (9.35) proporciona los deseados valores de los puntos estacionarios en función del parámetro de penalización r:

$$
\begin{aligned}
x_1(r) = \frac{77 - \sqrt{99(11 + 8r)}}{44} \\
x_2(r) = \frac{77 - \sqrt{11(11 + 8r)}}{44}
\end{aligned}
$$

donde las demás raíces han sido eliminadas por dar lugar a puntos infactibles. La figura 9.7 muestra las curvas de nivel de la función objetivo además de la región factible. Nótese que la solución óptima se alcanza en la frontera de la región factible, y que la trayectoria hacia el óptimo está contenida dentro de la región factible. Obsérvese además que

$$
\begin{aligned}
\lim_{r \to 0^+} x_1(r) &= 1 \\
\lim_{r \to 0^+} x_2(r) &= \frac{3}{2} \\
\lim_{r \to 0^+} r\phi(\mathbf{x}) &= 0
\end{aligned}
$$

■

Ejemplo 9.7 (método barrera). Considérese el problema: Minimizar

$$
Z = 9 - 8x_1 - 6x_2 - 4x_3 + 2x_1^2 + 2x_2^2 + x_3^2 + 2x_1x_2 + 2x_1x_3
$$

sujeto a

$$
\begin{aligned}
x_1 + x_2 + 2x_3 &\leq 3 \\
x_i &\geq 0, \quad i = 1, 2, 3
\end{aligned}
$$

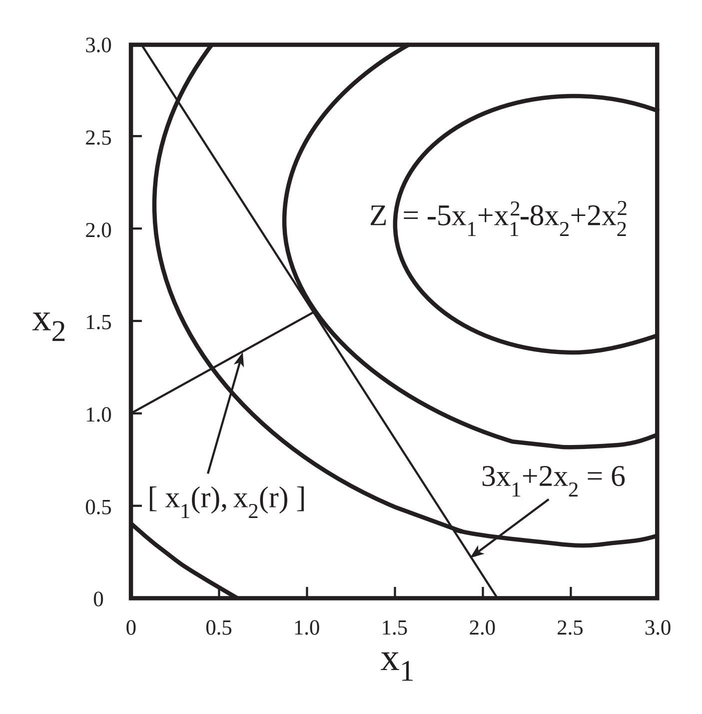


Figura 9.7: Ilustración del método barrera logarítmica.

Para calcular la solución de este problema mediante el método barrera se ha empleado el paquete MATLAB. Este programa emplea métodos quasi-Newton para resolver los problemas sin restricciones, y emplea la fórmula BFGS para actualizar las aproximaciones a la matriz hessiana. Para realizar las búsquedas lineales emplea un método de interpolación polinomial (cúbica).

La función barrera logarítmica no está definida para valores negativos de x. No obstante, el empleo de un método de barrera logarítmica puede dar lugar a puntos no factibles. Por tanto, debe considerarse algún mecanismo para detectar, prevenir y resolver dicha situación. En este ejemplo, y para las penalizaciones barrera logarítmica y barrera inversa, se pueden generar puntos infactibles. Para evitar esta posibilidad, se ha introducido la siguiente modificación en el término de penalización:

$$\hat{\phi}(\mathbf{g}) = \phi(\hat{\mathbf{g}}) \text{ donde } \hat{g}_i = \min\{g_i, -\varepsilon\}$$

para pequeños valores de la tolerancia ε.

Los resultados de la optimización numérica se resumen en la tabla 9.8.

■

Tabla 9.8: Resultados para el ejemplo 9.7

	t	r_t	$x_1^{(t)}$	$x_2^{(t)}$	$x_3^{(t)}$	$f(x^{(t)})$	$P(\mathbf{x}^{(t)}, r_t)$	$r_t\phi(g(\mathbf{x}^{(t)}))$
Penalización	1	1.00	1.123	0.775	0.598	0.703	7.415	6.712
inversa	2	10^{-1}	1.292	0.697	0.331	0.232	1.043	0.810
	3	10^{-2}	1.379	0.737	0.351	0.154	0.259	0.104
	4	10^{-3}	1.379	0.795	0.351	0.125	0.145	0.020
	5	10^{-4}	1.373	0.799	0.374	0.115	0.121	0.005
	6	10^{-5}	1.335	0.776	0.440	0.112	0.114	0.001
	7	10^{-6}	1.336	0.775	0.442	0.111	0.112	0.000
Penalización	1	1.00	1.164	0.732	0.314	0.354	2.415	2.061
logarítmica	2	10^{-1}	1.372	0.736	0.333	0.168	0.426	0.258
	3	10^{-2}	1.344	0.769	0.422	0.120	0.160	0.040
	4	10^{-3}	1.334	0.776	0.442	0.112	0.118	0.006
Solución	∞		$\frac{12}{9}$	$\frac{7}{9}$	$\frac{4}{9}$	$\frac{1}{9}$	$\frac{1}{9}$	0

9.2.3 Método del punto interior en programación lineal

Durante décadas el método simplex ha sido el método de resolución de los problemas de programación lineal. Si embargo, desde el punto de vista teórico, el tiempo de cálculo requerido por este método crece exponencialmente con el tamaño del problema, definido como el número de variables más el de restricciones. Este crecimiento exponencial corresponde al peor caso posible que no corresponde con lo que normalmente ocurre en las casos reales. Muchos investigadores han tratado de desarrollar algoritmos cuyos tiempos de cálculo tuviesen un crecimiento polinomial con el tamaño del problema. Todos los intentos realizados hasta el 1984 fallaron, ya fuese desde el punto de vista teórico o desde el punto de vista práctico. En 1984, Karmarkar [57] propuso un algoritmo cuya complejidad computacional es polinomial y que resultó altamente competitivo frente al método simplex para resolver problemas de programación lineal de gran tamaño. Trabajos pioneros en métodos de punto interior, anteriores al algoritmo de Karmarkar, son el de Frisch [45], Fiacco y McCormick [38], y Khachiyan [63]. El algoritmo de Karmarkar originó multitud de trabajos alrededor de su idea original que ha sido mejorada en muchos aspectos. Una de las más fructíferas variantes es el algoritmo barrera primal-dual introducido por Meggido [72] y explicado en este capítulo siguiendo el de Torres y Quintana [90] y Medina et al. [71]. Información más detallada sobre el algoritmo del punto interior primal-dual se encuentra en Wright [105].

Fundamentos

Esta sección desarrolla los elementos que sirven de base al algoritmo de barrera logarítmica primal-dual.

Considérese el siguiente PPL, expresado en forma estándar, como problema primal: Minimizar

$$Z = \mathbf{c}^T\mathbf{x} \tag{9.36}$$

sujeto a

$$\begin{array}{rcl} \mathbf{A}\mathbf{x} & = & \mathbf{b} \\ \mathbf{x} & \geq & \mathbf{0} \end{array} \tag{9.37}$$

donde $\mathbf{x}, \mathbf{c} \in \mathbb{R}^n$, $\mathbf{b} \in \mathbb{R}^m$ y $\mathbf{A} \in \mathbb{R}^m \times \mathbb{R}^n$.

El problema dual del anterior, es el siguiente: Maximizar

$$Z = \mathbf{b}^T\mathbf{y} \tag{9.38}$$

sujeto a

$$\mathbf{A}^T\mathbf{y} \quad \leq \quad \mathbf{c} \tag{9.39}$$

y convirtiendo las desigualdades en igualdades y empleando variables de holgura, el problema anterior tiene la forma siguiente: Maximizar

$$Z = \mathbf{b}^T\mathbf{y} \tag{9.40}$$

sujeto a

$$\begin{array}{rcl} \mathbf{A}^T\mathbf{y} + \mathbf{z} & = & \mathbf{c} \\ \mathbf{z} & \geq & \mathbf{0} \end{array} \tag{9.41}$$

Las variables duales son $\mathbf{y} \in \mathbb{R}^m$ y $\mathbf{z} \in \mathbb{R}^n$.

Para eliminar las restricciones de no negatividad, se emplea la función barrera logarítmica; entonces se aborda el PPNL: Maximizar

$$Z = \mathbf{b}^T\mathbf{y} + \mu \sum_{j=1}^{n} \ln z_j \tag{9.42}$$

sujeto a

$$\mathbf{A}^T\mathbf{y} + \mathbf{z} \quad = \quad \mathbf{c} \tag{9.43}$$

Nótese que z_j nunca será cero para que la función de penalización esté bien definida.

Los métodos de punto interior resuelven el problema anterior para diferentes valores del parámetro μ. Este parámetro se define de manera tal que $\mu_0 > \mu_1 > \mu_2 > \ldots > \mu_\infty = 0$. El teorema siguiente muestra el porqué de esta elección.

Teorema 9.5 (convergencia). *La sucesión de parámetros $\{\mu_t\}_{t=0}^{\infty}$ genera una sucesión de problemas del tipo (9.42)–(9.43). La sucesión de soluciones a estos problemas, $\{x_t\}_{t=0}^{\infty}$, converge a la solución del problema (9.40)–(9.41).* ■

La demostración de este teorema puede encontrarse, por ejemplo, en Wright [105].

La función Lagrangiana del problema (9.42)-(9.43) tiene la forma

$$\mathcal{L}(\mathbf{x}, \mathbf{y}, \mathbf{z}, \mu) = \mathbf{b}^T\mathbf{y} + \mu \sum_{j=1}^{m} \ln z_j - \mathbf{x}^T(\mathbf{A}^T\mathbf{y} + \mathbf{z} - \mathbf{c}) \qquad (9.44)$$

Nótese que los multiplicadores de Lagrange, $\mathbf{x}$, son las variables del problema original (primal).

Empleando el anterior Lagrangiano, las condiciones de optimalidad de primer orden del problema (9.42)–(9.43) son

$$\left\{ \begin{array}{l} \nabla_{\mathbf{x}}\mathcal{L}(\mathbf{x}, \mathbf{y}, \mathbf{z}, \mu) = \mathbf{A}^T\mathbf{y} + \mathbf{z} - \mathbf{c} = \mathbf{0} \\ \nabla_{\mathbf{y}}\mathcal{L}(\mathbf{x}, \mathbf{y}, \mathbf{z}, \mu) = \mathbf{A}\mathbf{x} - \mathbf{b} = \mathbf{0} \\ \nabla_{\mathbf{z}}\mathcal{L}(\mathbf{x}, \mathbf{y}, \mathbf{z}, \mu) = \mathbf{X}\mathbf{Z}\mathbf{e} - \mu\mathbf{e} = \mathbf{0} \end{array} \right. \qquad (9.45)$$

donde

$$\begin{array}{rcl} \mathbf{X} & \equiv & \text{diag}(x_1, x_2, \ldots, x_n) \\ \mathbf{Z} & \equiv & \text{diag}(z_1, z_2, \ldots, z_m) \\ \mathbf{e} & = & (1, 1, \ldots, 1)^T \end{array} \qquad (9.46)$$

y la dimensión de $\mathbf{e}$ es $n \times 1$.

Esta notación compacta que emplea matrices diagonales se ilustra en el ejemplo 9.8.

Ejemplo 9.8 (ejemplo ilustrativo). Considérese el problema primal: Minimizar

$$Z = \begin{pmatrix} -3 & -5 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{pmatrix} \qquad (9.47)$$

sujeto a

$$\begin{pmatrix} 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 3 & 2 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{pmatrix} = \begin{pmatrix} 4 \\ 6 \\ 18 \end{pmatrix} \qquad (9.48)$$

$$x_1, x_2, x_3, x_4, x_5 \geq 0 \qquad (9.49)$$

La región factible de este problema, expresada en forma canónica, es la dada en la figura 9.8.

En su problema dual se maximiza

$$Z = \begin{pmatrix} 4 & 6 & 18 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} \qquad (9.50)$$

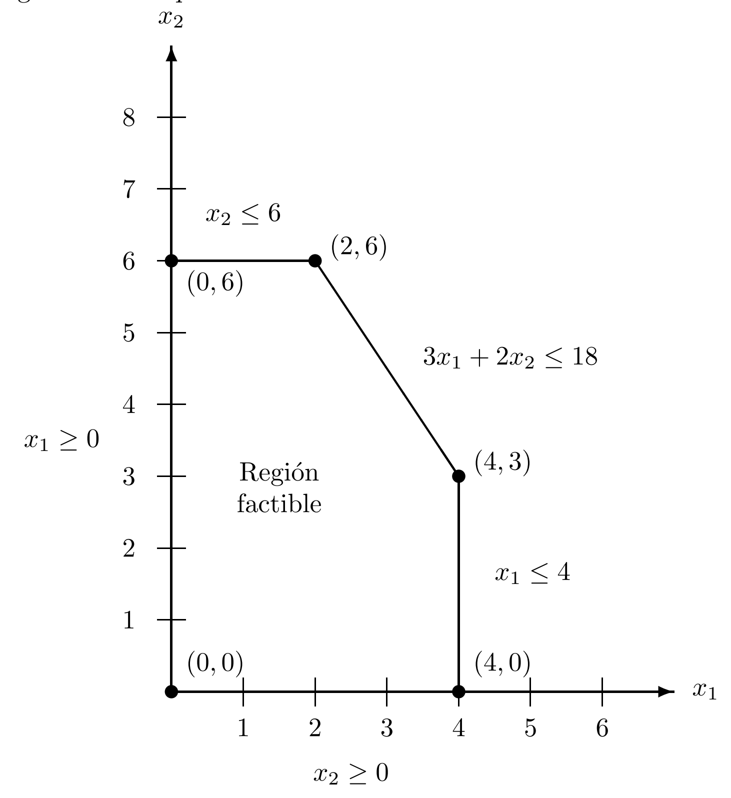


Figura 9.8: Ejemplo en forma canónica.

sujeto a

$$\begin{pmatrix} 1 & 0 & 3 \\ 0 & 1 & 2 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} + \begin{pmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \\ z_5 \end{pmatrix} = \begin{pmatrix} -3 \\ 5 \\ 0 \\ 0 \\ 0 \end{pmatrix} \tag{9.51}$$

$$z_1, z_2, z_3, z_4, z_5 \geq 0 \tag{9.52}$$

La función Lagrangiana es

$$\mathcal{L}(\mathbf{x}, \mathbf{y}, \mathbf{z}, \mu) = \begin{pmatrix} 4 & 6 & 18 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} + \mu \sum_{j=1}^{5} \ln z_j$$

$$-\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{pmatrix}^T \left\{ \begin{pmatrix} 1 & 0 & 3 \\ 0 & 1 & 2 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} + \begin{pmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \\ z_5 \end{pmatrix} - \begin{pmatrix} -3 \\ 5 \\ 0 \\ 0 \\ 0 \end{pmatrix} \right\} \quad (9.53)$$

Las condiciones de optimalidad (ecuaciones (9.45)) son

$$\begin{pmatrix} 1 & 0 & 3 \\ 0 & 1 & 2 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} + \begin{pmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \\ z_5 \end{pmatrix} - \begin{pmatrix} -3 \\ 5 \\ 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad (9.54)$$

$$\begin{pmatrix} 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 3 & 2 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{pmatrix} - \begin{pmatrix} 4 \\ 6 \\ 18 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \quad (9.55)$$

$$\begin{pmatrix} x_1 & & & & \\ & x_2 & & & \\ & & x_3 & & \\ & & & x_4 & \\ & & & & x_5 \end{pmatrix} \begin{pmatrix} z_1 & & & & \\ & z_2 & & & \\ & & z_3 & & \\ & & & z_4 & \\ & & & & z_5 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} - \begin{pmatrix} \mu \\ \mu \\ \mu \\ \mu \\ \mu \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad (9.56)$$

■

Para resolver el sistema de ecuaciones (9.45) mediante el método de Newton, $\mathbf{x}$, $\mathbf{y}$, y $\mathbf{z}$ son sustituidos respectivamente por $\mathbf{x} + \Delta\mathbf{x}$, $\mathbf{y} + \Delta\mathbf{y}$, y $\mathbf{z} + \Delta\mathbf{z}$, y eliminando los términos de segundo orden, se obtiene el siguiente sistema en los incrementos de las variables:

$$\begin{array}{rcl} \mathbf{A}\Delta\mathbf{x} & = & \mathbf{0} \\ \mathbf{A}^T\Delta\mathbf{y} + \Delta\mathbf{z} & = & \mathbf{0} \\ \mathbf{Z}\Delta\mathbf{x} + \mathbf{X}\Delta\mathbf{z} & = & \mu\mathbf{e} - \mathbf{XZe} \end{array} \quad (9.57)$$

Las direcciones de búsqueda primal y dual se obtienen resolviendo (9.57) para las variables $\Delta\mathbf{y}$, $\Delta\mathbf{z}$, y $\Delta\mathbf{x}$:

$$\begin{array}{rcl} \Delta\mathbf{y} & = & -(\mathbf{AXZ}^{-1}\mathbf{A}^T)^{-1}\mathbf{AZ}^{-1}\mathbf{v}(\mu) \\ \Delta\mathbf{z} & = & -\mathbf{A}^T\Delta\mathbf{y} \\ \Delta\mathbf{x} & = & \mathbf{Z}^{-1}\mathbf{v}(\mu) - \mathbf{XZ}^{-1}\Delta\mathbf{z} \end{array} \quad (9.58)$$

donde $\mathbf{v}(\mu) = \mu\mathbf{e} - \mathbf{XZe}$.

Seguidamente, se realiza una iteración de Newton, incluyendo el cálculo de la longitud de paso, para obtener el nuevo punto

$$\begin{array}{rcl} \mathbf{x}^{t+1} & = & \mathbf{x}^t + \alpha_p \Delta\mathbf{x} \\ \mathbf{y}^{t+1} & = & \mathbf{y}^t + \alpha_d \Delta\mathbf{y} \\ \mathbf{z}^{t+1} & = & \mathbf{z}^t + \alpha_d \Delta\mathbf{z} \end{array} \qquad (9.59)$$

donde $0 \leq \alpha_p \leq 1$, y $0 \leq \alpha_d \leq 1$ son, respectivamente, las longitudes de paso primal y dual. El paso primal se aplica a la variable primal $\mathbf{x}$, mientras que el paso dual se aplica a las variables duales $\mathbf{y}$ y $\mathbf{z}$.

La selección de α_p y α_d se realiza de modo que $\mathbf{x}$ y $\mathbf{z}$ permanezcan estrictamente positivos. Nótese que $\mathbf{y}$ no ha de ser positivo. Para asegurar la positividad, se emplea el parámetro σ $(0 < \sigma < 1)$. Entonces

$$\begin{array}{l} \alpha_x = \min\left\{ \dfrac{-x_i}{\Delta x_i} \text{ tal que } \Delta x_i < -\delta \right\} \\ \alpha_p = \min\{1, \sigma\alpha_x\} \end{array} \qquad (9.60)$$

$$\begin{array}{l} \alpha_z = \min\left\{ \dfrac{-z_j}{\Delta z_j} \text{ tal que } \Delta z_j < -\delta \right\} \\ \alpha_d = \min\{1, \sigma\alpha_z\} \end{array} \qquad (9.61)$$

donde δ es un parámetro de tolerancia (por ejemplo, $\delta = 0.0001$) y como σ se suele tomar el valor 0.99995.

Nótese que estas limitaciones en la longitud de paso son importantes para valores muy pequeños (negativos) de los incrementos Δx_i y Δz_j.

Holgura dual

Por la factibilidad de $\mathbf{x}$ y $\mathbf{y}$, la holgura dual se calcula mediante la expresión

$$\mathbf{c}^T\mathbf{x} - \mathbf{b}^T\mathbf{y}. \qquad (9.62)$$

Esta holgura dual se emplea como una medida de la proximidad a la solución óptima.

Infactibilidad inicial

Si el punto inicial es infactible, aplicando el método de Newton al sistema (9.45) (condiciones de optimalidad de primer orden) se llega al sistema incremental siguiente

$$\begin{array}{rcl} \mathbf{Z}\Delta\mathbf{x} + \mathbf{X}\Delta\mathbf{z} & = & \mu\mathbf{e} - \mathbf{XZe} \\ \mathbf{A}\Delta\mathbf{x} & = & \mathbf{b} - \mathbf{Ax} \\ \mathbf{A}^T\Delta\mathbf{y} + \Delta\mathbf{z} & = & \mathbf{c} - \mathbf{A}^T\mathbf{y} - \mathbf{z} \end{array} \qquad (9.63)$$

cuya solución es

$$
\begin{array}{rcl}
\Delta \mathbf{y} & = & -(\mathbf{AXZ}^{-1}\mathbf{A}^T)^{-1}(\mathbf{AZ}^{-1}\mathbf{v}(\mu) - \mathbf{AXZ}^{-1}r_d - r_p) \\
\Delta \mathbf{z} & = & -\mathbf{A}^T\Delta \mathbf{y} + r_d \\
\Delta \mathbf{x} & = & \mathbf{Z}^{-1}\mathbf{v}(\mu) - \mathbf{XZ}^{-1}\Delta \mathbf{z}
\end{array}
\tag{9.64}
$$

donde

$$
\begin{array}{rcl}
r_p & = & \mathbf{b} - \mathbf{Ax} \\
r_d & = & \mathbf{c} - \mathbf{A}^T\mathbf{y} - \mathbf{z}
\end{array}
\tag{9.65}
$$

son respectivamente los residuos primal y dual.

Si el punto inicial no es factible, la factibilidad y la optimalidad se alcanzan simultáneamente a medida que el algoritmo del punto interior progresa; en otras palabras, los residuos y la holgura dual tienden a cero simultáneamente. Si, por otro lado, el punto inicial es factible, el sistema (9.63) se reduce al sistema (9.45).

Actualización del parámetro de penalización

Una posible elección de μ es la siguiente. De las condiciones de optimalidad (9.45), se obtiene la siguiente igualdad:

$$
\mu = \frac{\mathbf{z}^T\mathbf{x}}{n}
\tag{9.66}
$$

Para alcanzar un "camino central", que evite una prematura aproximación a la frontera, se introduce el parámetro ρ. Entonces

$$
\mu = \rho\frac{\mathbf{z}^T\mathbf{x}}{n}
\tag{9.67}
$$

Los valores del parámetro ρ se ajustan experimentalmente mediante prueba y error. Una elección razonable es la dada en [100]

$$
\rho = \left\{ \begin{array}{lll} 0.1 & \text{si} & \mathbf{c}^T\mathbf{x} > \mathbf{b}^T\mathbf{y} \\ 0.2 & \text{si} & \mathbf{c}^T\mathbf{x} < \mathbf{b}^T\mathbf{y} \end{array} \right.
\tag{9.68}
$$

Si $\mathbf{c}^T\mathbf{x} = \mathbf{b}^T\mathbf{y}$, la solución óptima ha sido encontrada.

Criterio de parada

El algoritmo termina cuando la holgura dual es suficientemente pequeña, esto es, cuando

$$
\frac{|\mathbf{c}^T\mathbf{x}^t| - |\mathbf{b}^T\mathbf{y}^t|}{\max\{1, |\mathbf{c}^T\mathbf{x}^t|\}} < \varepsilon
\tag{9.69}
$$

donde ε es una tolerancia unitaria.

El numerador de la anterior fracción representa la holgura dual, mientras que el denominador es el máximo entre 1 y el valor de la función objetivo del problema primal. Esta elección evita dividir por cero.

Cotas

Considérese que el problema primal original contiene acotaciones superiores e inferiores en las variables $\mathbf{x}$, como el problema: Minimizar

$$Z = \mathbf{c}^T\mathbf{x} \tag{9.70}$$

sujeto a

$$\begin{array}{c} A\mathbf{x} = \mathbf{b} \\ \mathbf{l} \le \mathbf{x} \le \mathbf{u} \end{array} \tag{9.71}$$

donde algunas de las componentes del vector $\mathbf{u}$ pueden ser infinitas y algunas de $\mathbf{l}$ pueden ser cero.

Empleando las variables de holgura $\mathbf{s}$ y $\mathbf{v}$, el problema anterior se puede escribir como sigue: Minimizar

$$Z = \mathbf{c}^T\mathbf{x} \tag{9.72}$$

sujeto a

$$\begin{array}{rcl} \mathbf{Ax} & = & \mathbf{b} \\ \mathbf{x}+\mathbf{s} & = & \mathbf{u} \\ \mathbf{x}-\mathbf{v} & = & \mathbf{l} \\ \mathbf{s} & \ge & \mathbf{0} \\ \mathbf{v} & \ge & \mathbf{0} \end{array} \tag{9.73}$$

y empleando el método de la barrera logarítmica, el problema anterior es ahora la minimización de

$$Z = \mathbf{c}^T\mathbf{x} - \mu\left(\sum_{i=1}^{n}(\ln s_i + \ln v_i)\right) \tag{9.74}$$

sujeto a

$$\begin{array}{rcl} \mathbf{Ax} & = & \mathbf{b} \\ \mathbf{x}+\mathbf{s} & = & \mathbf{u} \\ \mathbf{x}-\mathbf{v} & = & \mathbf{l} \end{array} \tag{9.75}$$

y su Lagrangiano tiene la forma

$$\begin{array}{rcl} \mathcal{L}(\mathbf{x},\mathbf{y},\mathbf{z},\mu) & = & \mathbf{c}^T\mathbf{x} - \mu\left(\sum_{i=1}^{n}(\ln s_i + \ln v_i)\right) \\ & & -\mathbf{y}^T(A\mathbf{x}-\mathbf{b}) + \mathbf{w}^T(\mathbf{x}+\mathbf{s}-\mathbf{u}) - \mathbf{z}^T(\mathbf{x}-\mathbf{v}-\mathbf{l}) \end{array} \tag{9.76}$$

donde las variables duales asociadas a las restricciones $\mathbf{Ax} = \mathbf{b}$ son $\mathbf{y}$, las variables duales asociadas a las restricciones $\mathbf{x}+\mathbf{s} = \mathbf{u}$ son $\mathbf{w}$, y las variables duales asociadas a las restricciones $\mathbf{x}-\mathbf{v} = \mathbf{l}$ son $\mathbf{z}$.

Las condiciones de optimalidad resultan

$$\left\{\begin{array}{rcl} \nabla_{\mathbf{x}}\mathcal{L}(\mathbf{x},\mathbf{y},\mathbf{z},\mu) & = & \mathbf{A}^T\mathbf{y}+\mathbf{z}-\mathbf{w}-\mathbf{c}=\mathbf{0} \\ \nabla_{\mathbf{y}}\mathcal{L}(\mathbf{x},\mathbf{y},\mathbf{z},\mu) & = & \mathbf{Ax}-\mathbf{b}=\mathbf{0} \\ \nabla_{\mathbf{z}}\mathcal{L}(\mathbf{x},\mathbf{y},\mathbf{z},\mu) & = & \mathbf{x}-\mathbf{v}-\mathbf{l}=\mathbf{0} \\ \nabla_{\mathbf{s}}\mathcal{L}(\mathbf{x},\mathbf{y},\mathbf{z},\mu) & = & \mathbf{SWe}-\mu\mathbf{e}=\mathbf{0} \\ \nabla_{\mathbf{v}}\mathcal{L}(\mathbf{x},\mathbf{y},\mathbf{z},\mu) & = & \mathbf{VZe}-\mu\mathbf{e}=\mathbf{0} \\ \nabla_{\mathbf{w}}\mathcal{L}(\mathbf{x},\mathbf{y},\mathbf{z},\mu) & = & \mathbf{x}+\mathbf{s}-\mathbf{u}=\mathbf{0} \end{array}\right. \tag{9.77}$$

Para el caso usual de $\mathbf{l} = \mathbf{0}$, lo que implica $\mathbf{x} = \mathbf{v}$, estas condiciones pasan a ser

$$\begin{array}{rcl} \mathbf{A}\mathbf{x} & = & \mathbf{b} \\ \mathbf{x} + \mathbf{s} & = & \mathbf{u} \\ \mathbf{A}^T\mathbf{y} + \mathbf{z} - \mathbf{w} & = & \mathbf{c} \\ \mathbf{X}\mathbf{Z}\mathbf{e} & = & \mu\mathbf{e} \\ \mathbf{S}\mathbf{W}\mathbf{e} & = & \mu\mathbf{e} \end{array} \tag{9.78}$$

La dirección de búsqueda resultante de aplicar el método de Newton al sistema anterior es

$$\begin{array}{rcl} \Delta\mathbf{y} & = & (\mathbf{A}\Theta\mathbf{A})^{-1}\left[(\mathbf{b} - \mathbf{A}\mathbf{x}) + \mathbf{A}\Theta((\mathbf{c} - \mathbf{A}^T\mathbf{y} - \mathbf{z} + \mathbf{w}) + \boldsymbol{\rho}(\mu))\right] \\ \Delta\mathbf{x} & = & \Theta\left[\mathbf{A}^T\Delta\mathbf{y} - \boldsymbol{\rho}(\mu) - (\mathbf{c} - \mathbf{A}^T\mathbf{y} - \mathbf{z} + \mathbf{w})\right] \\ \Delta\mathbf{z} & = & \mu X^{-1}\mathbf{e} - \mathbf{Z}\mathbf{e} - \mathbf{X}^{-1}\mathbf{Z}\Delta\mathbf{x} \\ \Delta\mathbf{w} & = & \mu\mathbf{S}^{-1}\mathbf{e} - \mathbf{W}\mathbf{e} + \mathbf{S}^{-1}\mathbf{W}\Delta\mathbf{x} \\ \Delta\mathbf{s} & = & -\Delta\mathbf{x} \end{array} \tag{9.79}$$

donde

$$\begin{array}{rcl} \Theta & = & (\mathbf{X}^{-1}\mathbf{Z} + \mathbf{S}^{-1}\mathbf{W})^{-1} \\ \boldsymbol{\rho}(\mu) & = & \mu(\mathbf{S}^{-1} - \mathbf{X}^{-1})\mathbf{e} - (\mathbf{W} - \mathbf{Z})\mathbf{e} \end{array} \tag{9.80}$$

Nótese que aunque las ecuaciones anteriores llegan a ser ligeramente más complicadas que las correspondientes al caso de no incluir cotas, el esfuerzo computacional es prácticamente el mismo en ambos casos.

Selección del punto inicial

Si no se dispone de un punto inicial factible, el siguiente algoritmo heurístico, debido a Vanderbei [100], suele proporcionar un buen punto de inicio.

Primeramente, se calcula el vector $\hat{\mathbf{x}}$ mediante la fórmula

$$\hat{x}_j = \frac{1}{\|A_j\|_2 + 1} \tag{9.81}$$

donde A_i es la j-ésima columna de la matriz de restricciones $\mathbf{A}$, y $\|\cdot\|_2$ es la norma euclídea.

Entonces, el escalar β se calcula mediante

$$\beta = \frac{\|\mathbf{b}\|_2 + 1}{\|\mathbf{A}\hat{\mathbf{x}}\|_2 + 1} \tag{9.82}$$

El vector inicial $\mathbf{x}^{(0)}$ se calcula finalmente como

$$\mathbf{x}^{(0)} = 10\,\beta\,\hat{\mathbf{x}}. \tag{9.83}$$

El vector de variables duales $\mathbf{y}$ se hace $\mathbf{0}$. El vector $\mathbf{z}$ de variables duales se inicia en función del vector $\mathbf{x}$ del siguiente modo.

Si x_j tiene una cota superior finita, entonces

$$z_j^0 = \begin{cases} 1 & \text{si } c_j < 0 \\ c_j + 1 & \text{en otro caso} \end{cases} \tag{9.84}$$

Si, por otro lado, x_j no está acotado superiormente, entonces

$$z_j^0 = \begin{cases} 1 & \text{si } c_j < 1 \\ c_j & \text{en otro caso} \end{cases} \tag{9.85}$$

Algoritmo 9.6 (Método barrera logarítmica primal–dual).

Entrada. El problema de programación lineal (9.36)-(9.37).

Salida. Su solución o un mensaje indicando que es no acotado o no tiene solución.

Paso 0. Se toma $t = 0$ y se selecciona un punto inicial $(\mathbf{x}^t, \mathbf{y}^t, \mu_t)$, empleando la ecuación (9.83). Este punto inicial puede ser factible o infactible para el problema primal y/o dual. Se calcula el vector de holgura dual $\mathbf{z}^t = \mathbf{c} - \mathbf{A}^T\mathbf{y}^t$.

Paso 1. Se calculan las matrices diagonales $\mathbf{X}^t = \text{diag}(x_1^t, x_2^t, \ldots, x_n^t)$ y $\mathbf{Z}^t = \text{diag}(z_1^t, z_2^t, \ldots, z_m^t)$.

Paso 2. Se calcula el vector $\mathbf{v}(\mu_t) = \mu_t\mathbf{e} - \mathbf{X}^t\mathbf{Z}^t\mathbf{e}$, y los vectores residuos $r_p^t = \mathbf{A}\mathbf{x}^t - \mathbf{b}$ y $r_d^t = \mathbf{c} - \mathbf{A}^T\mathbf{y}^t - \mathbf{z}^t$. Si el punto actual es factible para los problemas primal y dual, los residuos son cero.

Paso 3. Se resuelve el sistema de Newton para obtener las direcciones de búsqueda $\Delta\mathbf{y}^t$, $\Delta\mathbf{z}^t$, y $\Delta\mathbf{x}^t$, empleando las ecuaciones (9.64).

Paso 4. Se calculan las longitudes de paso α_p^t y α_d^t, empleando las ecuaciones (9.60) y (9.61).

Paso 5. Se actualizan las variables duales y primales, empleando las ecuaciones (9.59).

Paso 6. Si la holgura dual es suficientemente pequeña (ecuación (9.69)), se *para*, pues se ha alcanzado la solución. En caso contrario, se hace $t \leftarrow t + 1$ y se continúa.

Paso 7. Se actualiza el parámetro de penalización μ_t (ecuaciones (9.67) y (9.68)) y se va al paso 1.

■

Ejercicios

9.1 Encuéntrese el mínimo de $f(x) = \exp(x - 1) - x$ empleando cada uno de los siguientes métodos:

(a) Método quasi-Newton (secante)
(b) Método de Newton
(c) Mediante interpolación cuadrática
(d) Regla de Armijo

9.2 Repítase el ejercicio anterior con las funciones

$$f(x) = x^4 - 6x^2$$

y

$$f(x) = x \ln x$$

9.3 Las pérdidas de energía de un generador de potencia superconductor dependen de la presión y temperatura del fluido de refrigeración según la expresión

$$f(x_1, x_2) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2$$

donde $f(x_1, x_2)$ son las pérdidas, x_1 la presión, y x_2 la temperatura. Determinar las condiciones de trabajo que minimizan las pérdidas empleando los siguientes métodos:

(a) Método del gradiente
(b) Método de Newton
(c) Un método quasi-Newton
(d) Un método de direcciones conjugadas

9.4 Considérese el problema: Minimizar

$$Z = (x_1 - 2)^2 + 4(x_2 - 6)^2$$

sujeto a

$$\begin{array}{rcl} 12x_1 + 6(x_2 + 1)^2 & \leq & 4 \\ x_1, x_2 & \geq & 0. \end{array}$$

(a) Calcúlese la solución óptima empleando las condiciones de KKT.
(b) Empléese un método de direcciones conjugadas para resolver el problema sin restricciones comenzando desde el origen. Realícense al menos tres iteraciones.

9.5 Selecciónese un posible método para resolver el problema: Minimizar

$$Z = x_1^2 + 2x_2^2 + x_1x_2 - 4x_1 - 6x_2$$

sujeto a:

$$\begin{array}{rcl} x_1 + 2x_2 & \leq & 4 \\ x_1, x_2 & \geq & 0 \end{array}$$

y justifíquese esta selección.

9.6 Resuélvase el problema: Minimizar

$$Z = (x - 2)^2$$

sujeto a

$$x \leq 1$$

mediante cada uno de los siguiente métodos:

(a) Método de barrera logarítmica

(b) Método de barrera inversa

9.7 Considérese el problema: Minimizar

$$Z = -2x_1 + 8x_2 + x_1x_2 + x_1^2 + x_1^2 + 6x_2^2$$

sujeto a:

$$\begin{aligned} x_1 + 2x_2 &\leq 4 \\ 2x_1 + x_2 &\leq 5 \\ x_1, x_2 &\geq 0 \end{aligned}$$

(a) Encuéntrese un mínimo local del problema sin restricciones, y justifíquese que se trata de una solución óptima del problema con restricciones.

(b) Empléese un método de direcciones de descenso para resolver el problema sin restricciones empezando desde el punto $(0, 0)$. Justifíquese la elección del método y los resultados obtenidos.

(c) Resuélvase el problema (b) empleando un algoritmo de este capítulo. Realícense al menos dos iteraciones.

(d) Derívese el problema dual y resuélvase mediante el método del gradiente.

9.8 Considérese el problema: Minimizar

$$Z = x_1^2 + x_2^2$$

sujeto a

$$(x_1 - 2)^2 + (x_2 - 2)^2 \leq 1$$

Resuélvase este problema mediante cada uno de los siguientes métodos de penalización:

(a) Función de penalización valor absoluto

(b) Penalización cuadrática

9.9 Considérese el problema: Minimizar

$$-x_1 - x_2$$

sujeto a

$$\begin{aligned} \exp(x_1) + \exp(x_2) &\leq 20 \\ x_1, x_2 &\geq 0 \end{aligned}$$

(a) Derívese la solución óptima empleando las condiciones KKT.

(b) Resuélvase este problema empleando el método dual.

9.10 Considérese el problema estrictamente convexo: Minimizar

$$Z = \sum_{i=1}^{m} f_i(x_i)$$

sujeto a

$$\begin{aligned} \sum_{i=1}^{m} g_{ij}(x_i) &\leq 0, \ j = 1, \ldots, r \\ x_i &\in X_i, \ i = 1, \ldots, m \end{aligned}$$

donde $f_i : \mathbb{R}^{n_i} \to \mathbb{R}$ son funciones estrictamente convexas, x_i son las componentes de x, g_{ij} son funciones convexa en $\mathbb{R}^{n_i}$, y X_i son subconjuntos convexos y compactos de $\mathbb{R}^{n_i}$.

(a) Escríbase el problema dual.

(b) *(Algoritmo de descenso coordenado).* El método de descenso coordenado se usa para minimizar funciones diferenciables sin hacer uso de las derivadas. La función objetivo se minimiza a lo largo de cada coordenada iterativamente. El orden en que se eligen estas coordenadas puede ir cambiando a lo largo de la ejecución del algoritmo. En el caso más usual este orden es cíclico, esto es, dado $\mathbf{x}^{(t)}$, la i-ésima coordenada de $\mathbf{x}^{(t+1)}$ se determina resolviendo

$$x_i^{(t+1)} = \text{arg minimiza}_{\zeta \in \mathbb{R}} \ \ f(\mathbf{x}_1^{(t)}, \ldots, \mathbf{x}_{i-1}^{(t)}, \zeta, \mathbf{x}_{i+1}^{(t)}, \ldots, \mathbf{x}_n^{(t)})$$

¿Cómo se puede usar la estructura del problema dual para poder aplicar el algoritmo del descenso coordenado mediante computación paralela?

(c) Para funciones objetivo continuamente diferenciables, se puede probar que el límite de la sucesión generada es un punto estacionario, esto es, si $\{\mathbf{x}^{(t)}\}_{t \in T} \to \bar{\mathbf{x}}$, entonces $\nabla f(\bar{\mathbf{x}}) = 0$. Muéstrese en este caso que el algoritmo del descenso coordenado converge a la solución óptima del problema dual.

9.11 Resuélvase el problema de programación lineal: Minimizar

$$Z = x_1 - x_2$$

sujeto a:

$$\begin{array}{rcl} x_1 & \leq & 1 \\ x_2 & \leq & 1 \\ x_1, x_2 & \geq & 0 \end{array}$$

empleando el método del punto interior.

Parte III

Software

Capítulo 10

La herramienta GAMS

10.1 Introducción

GAMS [53] (General Algebraic Modeling System) es un lenguaje de programación que permite el modelado, análisis y resolución de diversos problemas de optimización. Aunque inicialmente el manejo y comprensión de sus estructuras requiere cierto esfuerzo, una vez entendidas se dispone de una herramienta muy versátil capaz de resolver problemas de programación matemática. A pesar de ser una magnífica herramienta, el lector debe ser consciente de las limitaciones impuestas por el estado del arte existente en programación matemática.

Otros lenguajes similares a GAMS son AMPL [41, 54] y AIMMS [13, 55]. Todos ellos presentan características análogas y, en general, no hay razón alguna para elegir uno u otro. En este libro se opta por GAMS dada la familiaridad de los autores con este lenguaje.

Entre las características más importantes de GAMS cabe destacar

1. Su capacidad para pasar de resolver problemas de pequeña dimensión (docenas de variables y restricciones) a problemas mayores (miles de variables y restricciones) sin variar el código sustancialmente. El manejo eficiente de sus índices permite escribir de manera compacta restricciones similares mediante una sola restricción.

2. Separa el proceso de modelado del proceso de resolución del problema. Así, el usuario de GAMS debe ser capaz de conseguir una formulación consistente del problema, y una vez la expresa en la notación de GAMS, este lenguaje hace uso de alguno de los optimizadores disponibles para obtener su solución. De esta manera, el usuario sólo ha de centrarse en obtener un modelo del problema y puede ignorar el funcionamiento interno del algoritmo que se necesita para resolverlo. La separación de estas dos tareas permite cambiar el modelo para mejorarlo o completarlo cómodamente.

3. La forma en que GAMS representa un problema de optimización coincide, prácticamente, con la descripción matemática de ese problema. Por

tanto, el código GAMS es sencillo de comprender para aquellos lectores familiarizados con la optimización.

4. Además, GAMS proporciona los mecanismos necesarios para resolver problemas de optimización con estructuras similares, como son aquellos que se derivan de las técnicas de descomposición.

El usuario de GAMS debe ser cuidadoso con las reglas “gramaticales” de GAMS. El incumplimiento de una sola de ellas puede provocar muchos errores de compilación.

Entre la bibliografía de este lenguaje de programación cabe destacar el manual de GAMS [53], cuyo segundo capítulo ofrece un resumen con las características principales para empezar a programar en este lenguaje, y el artículo [24], que proporciona un enfoque ingenieril de GAMS.

En lo que sigue, se analizará un ejemplo para mostrar las características básicas de este lenguaje.

10.2 Ejemplo ilustrativo

Se ha elegido el problema del transporte para resaltar las características principales de GAMS. Este problema se describe a continuación.

Considérese un conjunto de consumidores y productores. Conocida (1) la demanda de cada consumidor, (2) la producción máxima de los productores, y (3) el coste del transporte de los productos desde los productores hasta los consumidores, el problema del transporte consiste en determinar la cantidad de producto a transportar de forma que el coste sea mínimo.

La formulación del problema anterior de programación matemática es como sigue. Minimizar

$$\sum_i \sum_j c_{ij} x_{ij}$$

sujeto a

$$\begin{array}{rcl} \sum_j x_{ij} & \leq & a_i, \ \ \forall i \\ \sum_i x_{ij} & \geq & b_j, \ \ \forall j \\ x_{ij} & \geq & 0, \ \ \forall i, j \end{array}$$

donde los subíndices son

i: para los productores

j: para los consumidores

Los datos son

a_i: la producción máxima del productor i en toneladas

b_j: la demanda del consumidor j en toneladas

Tabla 10.1: Distancias en kilómetros entre productores y consumidores

	Consumidores		
Productores	m_1	m_2	m_3
p1	2.0	1.6	1.8
p2	2.5	1.2	1.4

c_{ij}: el coste de transporte desde el productor i al consumidor j en euros por tonelada y kilómetro

Las variables de decisión son

x_{ij}: la cantidad de producto que se transporta desde el productor i hasta el consumidor j en toneladas

Considérense dos productores y dos consumidores. Las producciones máximas son 300 y 500 toneladas. Las demandas son de 100, 200 y 300 toneladas. Las distancias en kilómetros entre productores y consumidores vienen dadas en la Tabla 10.1.

El coste de transporte en euros es de 0.09 por tonelada y kilómetro.

En GAMS, el problema anterior se codifica y almacena en un fichero cuya extensión por defecto es '.gms'. El contenido de este fichero es el siguiente (el lector no debe tratar de comprender todo el código en esta primera aproximación, ya que a lo largo de este capítulo se explicarán en detalle las reglas del lenguaje; basta con adquirir una visión global de este fichero de entrada GAMS):

```
$Title  El Problema de Transporte  * Un ejemplo de transporte sencillo

  Sets
       i   productores   / p1, p2 /
       j   consumidores   / m1*m3 /;

  Table d(i,j)  distancia en kilómetros
                m1        m2         m3
      p1        2.0       1.6        1.8
      p2        2.5       1.2        1.4;

  Scalar f  coste del transporte (euros por tonelada y km)  /0.09/;

  Parameters
       a(i)  produccion maxima del productor i en toneladas
         /    p1    300
              p2    500  /
       b(j)  demanda del consumidor j en toneladas
         /    m1    100
              m2    200
              m3    300  /
```

```
     c(i,j)  coste de transporte en euros por tonelada;

c(i,j) = f * d(i,j);

Variables
     x(i,j)  cantidad de producto transportada entre i y j en toneladas
     z       coste total del transporte en euros;

Positive Variable x;

Equations
     coste       funcion objetivo
     maximo(i)   cumplimiento de maxima produccion del productor i
     demanda(j)  cumplimiento de la demanda del consumidor j;

coste ..        z  =e=  sum((i,j), c(i,j)*x(i,j));
maximo(i) ..    sum(j, x(i,j))  =l=  a(i);
demanda(j) ..   sum(i, x(i,j))  =g=  b(j);

Model transporte /all/;

Solve transporte using lp minimizing z;

Display x.l;
```

El lector puede comprobar con este ejemplo la similitud entre la formulación matemática y el código GAMS; concretamente, en la parte donde se definen las restricciones (la definición comienza en `coste ..`).

En la primera línea del código anterior se da nombre al problema, y en la segunda se escribe un texto a modo de comentario. En GAMS, cualquier línea que comienza con el símbolo "$*$" (asterisco) se interpreta como un comentario.

El comando `Sets` se emplea para definir dos estructuras típicas de GAMS: los índices, usados posteriormente para recorrer los vectores del problema, y los rangos de valores entre los que pueden variar estos índices. En el ejemplo propuesto, el índice `i` se refiere a los posibles productores `p1` y `p2`. De igual manera, el índice `j` se refiere a los posibles consumidores de `m1` a (representado por un asterisco) `m3`.

El comando `Table` permite definir matrices de datos (como puede ser la matriz de la Tabla 10.1 en el ejemplo). Obsérvese la facilidad con que se pueden representar las tablas en GAMS.

El comando `Scalar` es necesario en la declaración y asignación de escalares (como el escalar `f` en el ejemplo).

El comando `Parameters` se utiliza para declaración y asignación de vectores de datos. En este ejemplo se emplean dos vectores: `a(i)` y `b(j)`, que se refieren a la producción máxima, y a la demanda, respectivamente. La asignación mediante este comando se realiza para cada elemento del vector, por lo que es necesario especificar en primer lugar la posición del vector y en segundo, el valor a asignar.

Para declarar la matriz correspondiente al coste de transporte, `c(i,j)`, también se utiliza el comando `parameter`, sin embargo, la asignación de los elementos se realiza posteriormente como función de otros datos.

Las variables del problema de optimización se declaran mediante el comando `Variables`. Sólo cuando se completa la ejecución del código GAMS el optimizador determina el valor de estas variables. En GAMS se debe especificar el carácter de las variables, en este caso, se indica que `x(i,j)` es una variable positiva (no negativa).

El comando `Equations` permite indicar el nombre con que se referenciará a las restricciones del problema, incluida la función objetivo. Tras el comando, aparece el nombre en primer lugar; si es necesario indicar que existen varias restricciones con la misma estructura, se indica seguidamente mediante el índice correspondiente entre paréntesis. Una vez declaradas las restricciones con este comando, se pasa a su definición colocando el símbolo "`..`" y su formulación correspondiente en formato GAMS tras el nombre con el que se han declarado. Muchas funciones matemáticas se pueden emplear en la definición de las restricciones, como suma y multiplicación, entre otras.

El siguiente paso consiste en asociar un nombre al modelo y especificar qué conjunto de restricciones lo forman mediante el comando `Model`. En el ejemplo, el modelo se denomina `transporte`, y está formado por todas las restricciones previamente definidas (lo indica la palabra `all`).

Por último, se emplea el comando `Solve` para que GAMS llame al optimizador pertinente que finalmente resuelve el problema (en el ejemplo, la palabra `lp` indica que se debe usar un optimizador para programación lineal).

De manera opcional, se puede ordenar a GAMS que muestre el valor de los resultados más interesantes en un fichero de salida. Aquí, por ejemplo, el comando `Display x.l` tiene como objetivo mostrar el valor obtenido para el vector de variables `x(i,j)`.

Una vez que se ha escrito el código GAMS en el fichero de entrada, se compila; si no existen errores de sintaxis se resolverá el problema mediante el optimizador correspondiente y se escribirá la solución en un fichero de salida con extensión ".lst". En este fichero, aquellos valores de la solución que son cercanos a cero se representan mediante `EPS`, y aquellos que son cero se denotan mediante un punto.

Puesto que en el código del ejemplo propuesto se ha evitado cometer errores, en el fichero de salida se escribe la solución que ofrece el optimizador. Algunas de las partes que componen este fichero se muestran a continuación.

```
---- EQU MAXIMO      cumplimiento de maxima produccion del productor i
      LOWER     LEVEL     UPPER    MARGINAL
p1     -INF    100.000   300.000       .
p2     -INF    500.000   500.000    -0.036

---- EQU DEMANDA     cumplimiento de la demanda del consumidor j
      LOWER     LEVEL     UPPER    MARGINAL
m1   100.000   100.000     +INF      0.180
```

```
m2   200.000  200.000     +INF      0.144
m3   300.000  300.000     +INF      0.162
```

En esta parte del fichero se muestran los valores mínimo (`LOWER`), óptimo (`LEVEL`), máximo (`UPPER`), y el coste marginal (`MARGINAL`) asociado a las restricciones denominadas `MAXIMO` y `DEMANDA`.

Los valores máximos y mínimos que se asocian a una restricción dependen de su carácter. Así, si la restricción es del tipo "mayor o igual que" (como son las restricciones `SUPPLY`), el valor mínimo es $-\infty$ (`-INF` en el fichero de salida GAMS), y el valor máximo es el término independiente de esa restricción. Por otro lado, si la restricción es de "menor o igual que" (como son las restricciones `DEMANDA`), el valor mínimo es su término independiente, y el máximo es $+\infty$ (`+INF` en el fichero de salida GAMS). Si se tiene una restricción de igualdad, los valores máximo y mínimo coinciden con el valor de su término independiente.

El valor óptimo de una restricción se obtiene al evaluar su término dependiente una vez que se ha resuelto el problema.

Al igual que el valor óptimo, el coste marginal se obtiene como resultado de la optimización. Este valor representa la variación que experimenta el valor de la función objetivo frente a un cambio en el término independiente de la restricción (valor de su variable dual).

En el fichero de salida de GAMS también se presentan los valores que toman las variables de optimización tras la resolución del problema. A continuación, se muestra esta información.

```
---- VAR X cantidad de producto transportada entre i y j en toneladas
         LOWER     LEVEL     UPPER    MARGINAL
p1.m1      .      100.000     +INF       .
p1.m2      .         .        +INF      EPS
p1.m3      .         .        +INF       .
p2.m1      .         .        +INF     0.081
p2.m2      .      200.000     +INF       .
p2.m3      .      300.000     +INF       .

                       LOWER     LEVEL     UPPER    MARGINAL
---- VAR Z              -INF     77.400     +INF       .
  Z          coste total de transporte en euros
```

Se observa como para cada variable se obtiene su valor mínimo, óptimo, máximo y marginal. Si en el fichero de entrada no se especificó el límite inferior (superior) de la variable, entonces se considera que este valor es $-\infty$ ($+\infty$). Los valores óptimo y marginal de una variables se determinan una vez resuelto el problema.

Tabla 10.2: Algunos comandos de la herramienta GAMS

Comando	Objetivo
Set(s)	Declara un(os) conjunto(s) de índices y los propios índices de los vectores
Scalar(s)	Declara los escalares y, opcionalmente, les asigna un valor
Parameter(s)	Declara los vectores de datos y, opcionalmente, les asigna valores
Table(s)	Declara y asigna valores a las matrices de datos
Variable(s)	Declara las variables de optimización y su carácter proporcionándoles una cota superior e inferior
Equation(s)	Declara las restricciones y la función objetivo del problema
Model	Declara los modelos y las restricciones que los componen
Solve	Indica a GAMS que utilice un optimizador determinado para resolver el modelo
Display	Indica qué resultados deben ser presentados en el fichero de salida de GAMS

10.3 Características del lenguaje

En los siguientes apartados se describen los principales comandos de GAMS. En el manual de GAMS [53] se puede completar esta información.

Para resolver un problema usando GAMS es necesario crear un fichero de entrada donde se escribe la formulación del mismo con formato GAMS. Este fichero se divide en varios bloques y cada bloque se dedica a la representación de una parte del problema. En la Tabla 10.2 se muestran los distintos bloques (mediante los comandos asociados) de un fichero de entrada GAMS.

Antes de utilizar GAMS, es conveniente conocer sus reglas. A continuación, se describen las más importantes.

1. En GAMS, es indiferente el uso de mayúsculas o minúsculas.
2. Cada comando debe terminar con un punto y coma. El olvido de este separador de órdenes puede provocar muchos errores de compilación.
3. Los comandos pueden aparecer en cualquier lugar siempre que los datos y variables que se utilizan hayan sido declarados con anterioridad.
4. Como en cualquier otro lenguaje de programación, los identificadores que se utilicen para declarar datos o variables no pueden coincidir con las palabras reservadas de GAMS.
5. En algunos comandos de GAMS, el uso de la "s" final es indiferente. Así, se puede emplear indistintamente el comando `Set` o `Sets` para definir uno o varios conjuntos de índices.

6. Es posible escribir varios comandos en una misma línea siempre que estén separadas por punto y coma. El compilador considera sucesivos espacios en blanco como uno solo.

7. Un simple comando es válido para declarar o definir uno o varios elementos del mismo tipo. No hay necesidad de repetir el nombre del comando.

8. Para documentar el código GAMS es necesario incluir líneas de comentarios. El compilador de GAMS ignora cualquier línea cuya primera columna es un asterisco, considerándola un comentario. También es posible añadir un texto aclaratorio (opcionalmente entre comillas) tras algunos comandos de declaración en GAMS, como son: `set`, `scalar`, `parameter`, `table`, `equation`, y `model`. Este texto debe ayudar a comprender la utilidad del elemento.

9. La mayoría de los comandos son necesarios en la declaración de elementos y, opcionalmente, para asignarle valores. Declarar un elemento (escalar, vector, matriz, etc...) consiste en asignarle un identificador mediante cualquiera de los comandos declarativos (sets, scalar, parameter, table, variables, equations, y model). La declaración permite que un elemento pueda utilizarse en comandos posteriores.

10. Los identificadores usados en GAMS deben comenzar por una letra y pueden ir seguidos por hasta nueve caracteres alfanuméricos, no estando permitida la letra ñ, ni los caracteres especiales como los acentos (esto último tampoco está permitido en los textos explicativos).

10.3.1 Conjuntos

La palabra reservada `Set` o `Sets` se utiliza en GAMS para declarar un conjunto de índices y los propios índices que se emplearán para referirse a los componentes de un vector. Normalmente, la declaración del conjunto va acompañada de la relación de índices que lo componen. En el siguiente ejemplo se declaran los conjuntos `i` y `j`.

```
Sets
     i   productores   / p1, p2 /
     j   consumidores   / m1*m3 /;
```

El compilador de GAMS considera que el texto (`productores` y `consumidores`), que sigue al identificador del conjunto, es un comentario. Los elementos del conjunto, o posibles índices, se especifican entre dos símbolos "`/`" . Cuando los índices forman una secuencia ordenada de letras o números se puede utilizar un asterisco entre el primero y último elemento de la secuencia, evitando escribir cada uno de ellos. Por ejemplo, `/m1*m3/` equivale a `/m1, m2, m3/`. No obstante, GAMS no considera los índices como números sino como caracteres. El final de la declaración de los conjuntos puede terminar con un punto y coma.

Los conjuntos descritos anteriormente son conjuntos constantes o estáticos, es decir sus elementos no varían durante la ejecución del programa. En GAMS, también es posible definir conjuntos dinámicos que deben estar formados por elementos de conjuntos estáticos, previamente definidos. Los conjuntos dinámicos son útiles para definir subconjuntos cuyos elementos cambian en tiempo de ejecución. Dada la complejidad de estos conjuntos, se les dedica el apartado 10.3.12. Por otro lado, se puede establecer una aplicación que asocia elementos de conjuntos ya definidos. Como ejemplo, a continuación se representa una red de nudos definiendo una aplicación de dos elementos con las conexiones entre nudos.

```
Sets N       conjunto de nudos     /N1*N4/
     AP(N,N) conexion entre nudos  /N1.N2,N1.N3,N2.N4,N3.N4/
```

El código anterior define una red de 4 nudos y 4 conexiones. Para comprobar si dos nudos, n y n') están conectados se puede emplear el siguiente código (véase la Tabla 10.8 para más detalles de la sentencia `if`):

```
Alias(N,NP);
If(NP$AP(N,NP), ...; );
```

El comando `alias` es necesario en el caso anterior para referirse a distintos nudos dentro del mismo conjunto. Con este comando se consigue que se pueda usar indistintamente el identificador `N` o `NP` para referirse al conjunto de nudos (no se duplica el conjunto). Típicamente, los alias se usan en operaciones como sumas, productos, etc., donde se referencian dos o más índices del mismo conjunto y se necesita que varíen independientemente.

Para operar con elementos de conjuntos se definen en GAMS los siguientes operadores: `card`, `ord`, `+`, `-`, `++` y `--`, teniendo en cuenta que las operaciones sólo son válidas sobre conjuntos ordenados. Por conjunto ordenado se entiende aquel conjunto estático que está compuesto por una secuencia ordenada de índices. Existen distintas formas para conseguir que un conjunto estático sea ordenado:

- Si sus índices no pertenecen a ningún otro conjunto previamente declarado
- Si sus índices a pesar de pertenecer a otro conjunto previamente declarado y ordenado mantienen la misma posición relativa que en ese conjunto

A continuación se muestra un ejemplo con conjuntos ordenados y desordenados.

```
Sets A1 /Martes, Miercoles, Viernes, Jueves/
     A2 /Domingo, Jueves, Sabado, Viernes/
     A3 /Miercoles, Viernes, Sabado, Lunes/;
```

En este ejemplo, el conjunto `A1` está ordenado ya que sus elementos no pertenecen a ningún otro conjunto previamente definido. Sin embargo, el conjunto `A2` está desordenado porque la posición relativa entre el `Jueves` y el `Viernes` es distinta de su posición relativa en el conjunto `A1` definido con anterioridad. El último conjunto está ordenado porque la posición relativa de sus elementos es

Tabla 10.3: Operaciones sobre el conjunto ordenado `A1`

Operación	Resultado
`card(A1)`	4
`ord('Miercoles')`	2
`'Miercoles'+1`	Viernes
`'Jueves'++1`	Martes
`'Martes'--1`	Jueves

la misma que en el conjunto ordenado `A1`, y es independiente de que la posición relativa del `Viernes` y el `Sabado` en `A2` sea distinta, pues este conjunto está desordenado.

Para comprender mejor el funcionamiento de los operadores, en la Tabla 10.3 se muestra el resultado de las operaciones sobre el conjunto `A1`. El operador `card` devuelve el número de elementos que forman el conjunto (este operador es válido para conjuntos dinámicos o estáticos desordenados). El operador `ord` devuelve la posición que ocupa un elemento dentro del conjunto ordenado. Dado un elemento de un conjunto ordenado, es posible referirse al elemento inmediatamente anterior o posterior por medio de los operadores `+1` o `-1`. Si se usan estos operadores, no se considera que el conjunto es una lista circular de elementos, es decir, no existe el siguiente al último, ni el anterior al primero de sus elementos. Los operadores `++` y `--` permiten el tratamiento de conjuntos ordenados como listas circulares; por tanto, el siguiente del último es el primero, y el anterior al primero es el último elemento.

10.3.2 Escalares

En GAMS los escalares se declaran, y opcionalmente pueden iniciarse, usando el comando `Scalar` o `Scalars`. Se pueden incluir comentarios entre el identificador y el valor inicial, como se ilustra a continuación.

```
Scalar f  coste de transporte (euros por tonelada y km)  /0.09/;
```

Se declara el escalar `f` y se le asigna el valor inicial 0.09 que aparece entre barras "`/`". Con el punto y coma se pone fin a este comando declarativo.

10.3.3 Vectores y matrices de datos

Los comandos `Parameter` y `Table` se utilizan en GAMS para definir vectores de varias dimensiones. Estos comandos se pueden usar indistintamente, salvo para definir vectores de una sola dimensión. En estos casos, sólo se puede usar el comando `Parameter`. En este apartado, se explica, en primer lugar, el comando `Parameter`, en segundo, el comando `Table` y por último, se comparan ambos.

Para declarar vectores de datos se pueden utilizar las palabras reservadas `Parameter` o `Parameters` indistintamente. La asignación de valores iniciales es opcional, al igual que los comentarios aclaratorios.

```
Parameters
     a(i) produccion maxima del productor i en toneladas
       /    p1    300
            p2    500  /;
```

En el caso anterior, el parámetro `a(i)` tiene como índices aquellos que pertenecen al conjunto `i`. Para cada índice de este vector (`p1,p2`), se tiene un valor inicial distinto (`300,500`). La asignación de los valores inciales está comprendida entre dos barras "`/`". El punto y coma pone fin al comando. Para iniciar vectores en GAMS se deben tener presentes las siguientes reglas:

1. Las parejas de índice y valor inicial deben ir entre barras y separadas por comas o en distintas líneas.
2. Estas parejas pueden escribirse en cualquier orden.
3. El valor por defecto que GAMS asigna a los vectores de datos es cero. Por tanto, sólo es necesario especificar los valores distintos de cero.
4. Los índices deben pertenecer al conjunto con que se ha declarado el vector, si no se producirán errores en la compilación.

No siempre es necesario asignar un valor inicial a los vectores con este comando, la asignación se puede hacer posteriormente, como ocurre en el siguiente caso.

```
Parameter c(i,j)  coste del transporte en euros por tonelada;
          c(i,j) = f * d(i,j);
```

Obsérvese la capacidad que tiene GAMS para realizar una asignación escribiendo muy poco código. `c(i,j)` es una matriz que se ha definido como el producto de un escalar `f` ya definido, y otra matriz `d(i,j)`.

La asignación anterior es la versión compacta del siguiente código:

```
Parameter c(i,j)  coste del transporte en euros por tonelada
            /  p1.m1  0.180
               p2.m1  0.225
               p1.m2  0.144
               p2.m2  0.108
               p1.m3  0.162
               p2.m3  0.126  /;
```

Los vectores con más de dos dimensiones se pueden declarar indistintamente con los comandos `Parameter` o `Table`. Las palabras reservadas `Table` o `Tables` permiten definir un vector de dos o más dimensiones. Después del identificador del vector, en la sentencia de declaración pueden incluirse comentarios, como ocurre en el siguiente caso

```
Table d(i,j)  distancia en km
              m1       m2         m3
    p1        2.0      1.6        1.8
    p2        2.5      1.2        1.4;
```

En este ejemplo, la matriz de datos `d(i,j)` tiene como índices `i` y `j`. Los valores iniciales (`2.0, 1.6, 1.8, 2.5, 1.2, 1.4`) de la matriz se asignan a cada una de las posiciones (`p1.m1, p1.m2, p1.m3, p2.m1, p2.m2, p2.m3`).

Para concluir este apartado, se establece la equivalencia entre los dos comandos descritos para definir vectores. El siguiente código define el vector `c(i,j)` con el comando `Table`.

```
Table c(i,j)  coste de transporte en euros por tonelada
           m1     m2     m3
    p1     0.180  0.144  0.162
    p2     0.225  0.108  0.126;
```

Obsérvese que lo anterior no es la manera más compacta para representar la matriz `c(i,j)` en GAMS.

Para definir vectores de más de dos dimensiones considérese el siguiente ejemplo. Se quiere representar en un vector la cantidad de personas que coinciden en su religión, raza y nivel de estudios. El código para representar este vector en GAMS puede ser:

```
Sets re tipos de religion /re1*re3/
     sl niveles de estudios /sl1*sl2/
     ra tipos de razas /ra1*ra4/;

Table c(re,sl,ra)  cantidad de personas con coincidencias
                    ra1        ra2      ra3     ra4
    re1*re2.sl1      2          4       10       2
    re3.sl1          6          2        5       7
    re1*re3.sl2      3          9        7       8 ;

Display c;
```

Este código declara e inicia un vector de tres dimensiones de manera compacta (el funcionamiento del asterisco se explica en el apartado 10.3.9). Con el comando `Display` se escribirá la información siguiente en el fichero de salida GAMS.

```
 ----      11 PARAMETER C   cantidad de personas con coincidencias
               ra1         ra2         ra3         ra4
re1.sl1       2.000       4.000      10.000       2.000
re1.sl2       3.000       9.000       7.000       8.000
re2.sl1       2.000       4.000      10.000       2.000
re2.sl2       3.000       9.000       7.000       8.000
re3.sl1       6.000       2.000       5.000       7.000
re3.sl2       3.000       9.000       7.000       8.000
```

10.3.4 Reglas sobre las expresiones matemáticas en asignaciones

Se deben tener en cuenta las siguientes reglas a la hora de realizar asignaciones mediante el uso de expresiones matemáticas:

1. Es posible con una sola sentencia asignar valores a vectores y matrices, sin necesidad de recorrer mediante bucles cada uno de sus elementos. Por ejemplo, la siguiente asignación asigna un 3 a todos los elementos de la matriz c_{ij}.

```
c(i,j)=3;
```

2. Si se requiere asignar un valor a una posición concreta de la matriz c_{ij} se deben indicar los índices que la caracterizan entre comillas, como ocurre a continuación

```
c('p1','m1')=0.180;
```

3. Los valores asignados mediante los comandos Scalars, Parameters, y Tables pueden modificarse mediante una sentencia de asignación. Los valores que prevalecen son los correspondientes a la última asignación escrita.

4. Una expresión matemática puede contener cualquier operación y/o función matemática estándar. La Tabla 10.4 muestra algunas de las funciones matemáticas disponibles en GAMS. Por ejemplo, para representar un vector aleatorio $x_i, \forall i = 1 \cdots 10$ que sigue una distribución uniforme $U(0, 0.5)$, se puede usar el siguiente código:

```
Set I /I1*I10/; Parameter x(I);
x(I)=uniform(0,1)*0.5;
```

10.3.5 Variables

Las variables en GAMS se declaran de la siguiente manera:

```
Variables
     x(i,j)  cantidad transportada en toneladas
     z       coste de transporte en euros;
```

Las palabras reservadas `Variable` o `Variables` se utilizan indistintamente para declarar las variables a optimizar. Al declarar las variables de optimización es necesario especificar su dimensión. La variable que contendrá el valor de la función objetivo una vez resuelto el problema siempre debe ser declarada en GAMS (como la variable `z` en el ejemplo del transporte). Se pueden incluir comentarios tras el identificador de la variable. La declaración finaliza con un punto y coma.

Cuando se declaran las variables se puede, opcionalmente, indicar su naturaleza:

```
Positive Variable x;
```

Tabla 10.4: Algunas funciones matemáticas en GAMS

Función	Descripción
abs(x)	Valor absoluto de x
arctan(x)	Inversa de la función tangente (en radianes)
ceil(x)	El menor valor entero mayor o igual que x
cos(x)	Función coseno (x en radianes)
errorf(x)	Función de distribución de la distribución normal $N(0, 1)$ en x
exp(x)	Función exponencial
floor(x)	El mayor valor entero menor o igual que x
log(x)	El logaritmo natural de x
log10(x)	Logaritmo en base 10 de x
mapval(x)	Función aplicación
max(x_1,x_2,...)	Máximo de la lista
min(x_1,x_2,...)	Mínimo de la lista
mod(x,y)	Resto de la división de x entre y
normal(x,y)	Número aleatorio obtenido de una variable normal de media x y desviación típica y
power(x,y)	Función potencial x^y (donde y debe ser entero)
$x**y$	Función potencial x^y (donde x debe ser positivo)
round(x)	Redondeo de x al valor entero más cercano
round(x,y)	Redondeo de x con y posiciones decimales
sign(x)	Signo de x, 1 si positivo, -1 si negativo, y 0 si cero.
sin(x)	Función seno (en radianes)
sqr(x)	Cuadrado de x
sqrt(x)	Raíz cuadrada de x
trunc(x)	Equivale a sign(x) * floor(abs(x))
uniform(x,y)	Número aleatorio con distribución uniforme $U(x, y)$

Lo anterior establece una cota inferior para la variable `x(i,j)`, esto es, $x_{ij} \geq 0$. En este caso, no es necesario indicar de nuevo la dimensión de la variable, puesto que cuando se declaró ya se hizo.

Además, es posible declarar variables sin límites (opción por defecto) mediante el comando `Free`, variables binarias mediante `Binary`, enteras con `Integer`, o negativas con `Negative`. La variable que representa el valor de la función objetivo debe declararse sin límites usando el comando `Free`.

En la Tabla 10.5 se presentan los distintos tipos de variables y el intervalo de variación asociado.

El intervalo de variación que GAMS asigna por defecto se puede modificar mediante sufijos sobre las variables. Por ejemplo, la siguiente expresión matemática

$$2.0 \leq r \leq 5.0$$

Tabla 10.5: Tipos de variables e intervalo de variación

Tipo de variable	Intervalo	Intervalo por defecto
Binary	$\{0, 1\}$	$\{0, 1\}$
Free (default)	$(-\infty, \infty)$	$(-\infty, \infty)$
Integer	$\{0, 1, \ldots, n\}$	$\{0, 1, \ldots, 100\}$
Negative	$(-\infty, 0)$	$(-\infty, 0)$
Positive	$(0, \infty)$	$(0, \infty)$

se escribe en GAMS como

```
Positive variable r;
r.lo = 2.0;  r.up = 5.0;
```

donde el sufijo `lo` indica la cota inferior de la variable, mientras que el sufijo `up` denota su cota superior. Con estos sufijos se puede modificar, por ejemplo, el valor límite por defecto de una variable, como ocurre en el siguiente caso que se cambia el límite superior de una variable entera `i` de 100 a 1000:

```
Integer variable i; i.up=1000;
```

En el caso de querer fijar el valor de una variable, se puede asignar el mismo valor a su cota inferior y superior, o utilizar un nuevo sufijo `fx`. Por ejemplo, la expresión

$$y_i = 3.0, \quad \forall i$$

se escribe en GAMS como

```
y.fx(i) = 3.0;
```

Obsérvese que con una única sentencia GAMS se puede asignar el mismo valor a todos los elementos de un vector, independientemente de su dimensión.

Cuando se plantean problemas de programación no lineal es conveniente establecer el punto inicial de algunas variables. Por ejemplo, la siguiente asignación toma como punto inicial para la variable s_{ij} el valor 3.0 (no se congela la variable). Así la siguiente asignación

$$s_{ij} \leftarrow 3.0, \quad \forall i, \forall j$$

puede hacerse en GAMS como sigue

```
s.l(i,j) = 3.0;
```

el sufijo `l` obliga al optimizador a tomar el valor asignado como punto inicial.

10.3.6 Restricciones

Las palabras reservadas `Equation` o `Equations` se utilizan en GAMS para declarar restricciones y la función objetivo de problemas de optimización. Se pueden declarar tanto restricciones de igualdad (ecuaciones) como de desigualdad.

La declaración de restricciones en el ejemplo de transporte se muestra en las líneas 2–4 a continuación. Una vez declaradas, las restricciones se definen utilizando dos puntos entre su identificador y la definición (véanse las líneas 5–7). Para modelar las restricciones se utiliza la notación matemática estándar. La definición debe finalizar con un punto y coma.

```
Equations
     coste       funcion objetivo
     maximo(i)   cumplimiento de maxima produccion del productor i
     demanda(j)  cumplimiento de la demanda del consumidor j;

coste ..        z  =e=  sum((i,j), c(i,j)*x(i,j));
maximo(i) ..    sum(j, x(i,j))  =l=  a(i);
demanda(j) ..   sum(i, x(i,j))  =g=  b(j);
```

Obsérvese que el sumatorio `sum(i, x(i,j))` en GAMS indica $\sum_i x_{ij}$. Al igual que el sumatorio, la operación producto $\Pi_i x_{ij}$ se representa en GAMS como `prod(i,x(i,j))`.

La definición de las restricciones en GAMS debe incluir una de las siguientes secuencias:

- `=e=` para indicar que la restricción es de igualdad,
- `=l=` para indicar que la restricción es de “menor o igual que”
- `=g=` para indicar que la restricción es de “mayor o igual que”.

Si al declarar una restricción se incluyen conjuntos de índices tras su identificador significa que esta restricción representa en realidad un bloque de restricciones con idéntica estructura. El número de restricciones que representa el bloque viene dado por la dimensión de los conjuntos con que se declaren. Así, la siguiente definición de la restricción `maximo(i)`

```
maximo(i) ..    sum(j, x(i,j))  =l=  a(i);
```

equivale a las siguientes restricciones

```
maximo1 ..   sum(j,x('p1',j))  =l=  a('p1');
maximo2 ..   sum(j,x('p2',j))  =l=  a('p2');
```

El lector puede comprobar la capacidad de GAMS para compactar información.

10.3.7 Modelos

La palabra reservada `Model` se utiliza para identificar qué conjunto de las restricciones definidas forman parte de un modelo. En el ejemplo presentado, este comando asegura que se incluyan en el modelo denominado `transporte` todas (`all`) las restricciones definidas.

```
Model transporte /all/;
```

Lo anterior equivale al siguiente comando.

```
Model transporte /coste,maximo,demanda/;
```

Para un mismo conjunto de restricciones es posible definir varios modelos que contengan un subconjunto de ellas. Por ejemplo, el siguiente modelo `transporte1` no incluye las restricciones de `demanda`.

```
Model transporte1 "No incluye restriccion de demanda" /coste,maximo/;
```

Obsérvese que se han incluido comentarios entre comillas tras el identificador del modelo.

10.3.8 Resolución

La palabra reservada `Solve` se utiliza en GAMS para resolver el problema definido. En el ejemplo, este comando indica a GAMS que resuelva el modelo `transporte` usando un optimizador de programación lineal (`lp`) que minimize el valor de la variable `z`.

```
Solve transporte using lp minimizing z;
```

Además de la opción `lp`, utilizada en programación lineal, GAMS ofrece otras opciones como se indica en la Tabla 10.6.

Tabla 10.6: Opciones disponibles para la resolución de problemas en GAMS

Opción	Tipo de problema
lp	Programación lineal
nlp	Programación no lineal
dnlp	Programación no lineal con derivadas discontinuas
mip	Programación lineal entera-mixta
rmip	Programación lineal relajada entera-mixta
minlp	Programación no lineal entera-mixta
rminlp	Programación no lineal relajada entera-mixta
mcp	Problemas complementarios mixtos
mpec	Problemas matemáticos con restricciones de equilibrio
cns	Sistemas no lineales acotados

Una vez que se resuelve un problema con GAMS es posible extraer información de utilidad mediante algunos sufijos sobre el identificador del modelo resuelto. Entre los sufijos más importantes destacan: `modelstat` que informa sobre la calidad de la solución obtenida, `solvestat` que informa sobre el estado del optimizador al finalizar la resolución, y `resusd` que ofrece el tiempo de cálculo (expresado en segundos) que ha empleado el optimizador en resolver el problema. En el apartado 10.3.10 se presenta un ejemplo que usa el sufijo `resusd`. El siguiente ejemplo muestra el empleo del sufijo `modelstat`.

Tabla 10.7: Posibles valores para los sufijos `modelstat` y `solvestat`

Valor	`modelstat`	`solvestat`
1	Optimal (`lp`, `rmip`)	Normal completion
2	Locally optimal (`nlp`)	Iteration interrupt
3	Unbounded (`lp`, `rmip`, `nlp`)	Resource interrupt
4	Infeasible (`lp`, `rmip`)	Terminated by solver
5	Locally infeasible (`nlp`)	Evaluation error limit
6	Intermediate infeasible (`nlp`)	Error
7	Intermediate non-optimal (`nlp`)	-
8	Integer solution (`mip`)	-
9	Intermediate non-integer (`mip`)	-
10	Integer infeasible (`mip`)	-
11	–	-
12	Error unknown	-
13	Error no solution	-

```
If (transport.modelstat ne 1,
     Solve transport1 using lp minimizing z;
);
```

El código anterior comprueba en primer lugar si la solución obtenida para el modelo `transporte` es óptima mediante la condición `if` (se explica más adelante en este capítulo) y el sufijo `modelstat`. Si la solución no es óptima, se resuelve el modelo `transporte1` con menos restricciones. El sufijo `solvestat` se emplea de igual manera.

Según sea la naturaleza (lineal, no lineal, entera ...) del problema de optimización resuelto, el valor que devuelve GAMS al emplear los sufijos `solvestat` y `solvestat` es distinto. En la Tabla 10.7 se muestran los posibles valores asociados a estos dos sufijos. En esta tabla se aclara entre paréntesis cuál es la naturaleza de los problemas asociados a cada valor.

Los valores que indican que la solución es óptima son los siguientes. El valor 1 indica solución óptima si el problema resuelto es de programación lineal o programación lineal relajada entera-mixta. Para problemas no lineales y problemas enteros-mixtos este valor es 2 y 8, respectivamente, en la solución óptima.

El código 3 se utiliza para indicar que los modelos `lp`, `rmip`, o `nlp` no están acotados. Si ocurre esto, el usuario GAMS debe buscar en el fichero de salida la secuencia `UNBND` y determinar a qué variables afecta. Posteriormente, se debe modificar el fichero de entrada con nuevos límites sobre las variables afectadas.

Si los modelos son infactibles, el valor que devuelve el sufijo `modelstat` puede ser: 4 para modelos `lp` y `rmip`, 5 para modelos `nlp`, y 10 para modelos `mip`. En cualquiera de estos tres casos, el usuario debe buscar en el fichero de salida la secuencia `INFES` que aparece junto a las restricciones que no se han podido cumplir o a las variables cuyos valores no están comprendidos dentro de sus

límites. En estos casos, es necesario modificar los datos de entrada del problema o la estructura del problema hasta conseguir factibilidad en su solución.

Los valores anteriores (1, 2, 4, 5, 8 y 10) se obtienen al consultar la solución mediante el sufijo `modelstat`. Todos ellos se corresponden con el valor 1 para una consulta mediante el sufijo `modelstat`. Este valor indica que el optimizador no ha encontrado dificultades para obtener la solución o para determinar que ésta no existe. No obstante, si el problema es complejo y el optimizador necesita muchas iteraciones, el valor 2 indicará que el optimizador ha parado tras alcanzar el máximo permitido de iteraciones. En este caso, el usuario GAMS puede incrementar el límite máximo por defecto (1000 iteraciones) escribiendo el siguiente código antes de la aparición del comando de resolución `solve`.

```
Option iterlim=1e8;
```

En el caso de que el optimizador se detenga debido a los recursos limitados de la máquina, el valor obtenido con el sufijo `modelstat` es 3. Si 1000 segundos (valor por defecto) no son suficientes para resolver el problema se debe escribir el siguiente código antes de la aparición del comando de resolución `solve`.

```
Option reslim=1e10;
```

Por otro lado, se puede evitar que el optimizador alcance los límites anteriores cuando se resuelven modelos `mip` de gran dimensión. Para estos modelos, es posible elegir el grado de precisión de la solución entera en el óptimo. En el siguiente código aparecen las dos posibilidades que modifican dicha precisión (en ellas se muestran los valores que el optimizador entero toma por defecto):

```
Option optcr=0.1; ** O BIEN **
Option optca=0;
```

Los comandos anteriores establecen el criterio de parada que tiene en cuenta el correspondiente optimizador para modelos lineales enteros-mixtos. La primera de ellas, `optcr`, es un criterio para valores relativos, mientras que la segunda es un criterio para valores absolutos. El optimizador tomará el valor especificado por estos comandos para establecer una comparación entre cualquier solución entera y la mejor obtenida hasta entonces. Así, el valor 0.1 del primer criterio de parada indica que el optimizador se detiene cuando encuentra una solución entera cuya diferencia con la mejor solución posible sea menor que un 0.1%. En el segundo caso el optimizador se detiene cuando encuentra la solución óptima de manera exacta.

Si al consultar el estado del optimizador mediante el sufijo `solvestat` se obtiene un 4, indica que el optimizador no ha sido capaz de continuar con la resolución del problema. La razón por la que el optimizador para debe buscarse en el fichero de salida GAMS (fichero de extensión .lst).

Los valores 2, 3, y 4 para el sufijo `solvestat` se corresponden con los valores 6, 7, y 9 para el sufijo `modelstat`. Todos ellos indican que el optimizador ha alcanzado una solución intermedia (no óptima, pero factible) del problema. Los valores 6 y 7 están asociados a problemas de programación no lineal, pero se

diferencian en la "calidad" de la solución obtenida. El valor 6 indica que la solución no es factible, mientras que el valor 7 que el optimizador podría encontrar una solución factible si continuara la ejecución. Análogamente, el valor 9 indica que el optimizador no ha sido capaz de encontrar una solución factible para problemas de programación lineal entera-mixta en el tiempo disponible de ejecución.

El valor 5 para el sufijo `solvestat` indica que el optimizador necesita muchas evaluaciones de los términos no lineales en problemas de programación no lineal. En el fichero de salida se puede encontrar la ayuda necesaria para resolver este caso. Por último, los valores desde el 6 en adelante para el sufijo `solvestat` indican que el optimizador tiene grandes dificultades para resolver el problema. Estas dificultades pueden deberse a errores de preprocesado, fallos de iniciación, fallos del optimizador, errores internos del optimizador, errores de postprocesado, fallos del sistema o, simplemente, a errores no identificados.

10.3.9 La potencia del asterisco

En este apartado, se recopilan los distintos usos del asterisco en GAMS. Este símbolo puede usarse para:

- Hacer comentarios aclaratorios en el fichero de entrada. Aquellas líneas que contienen este símbolo en la primera columna son comentarios en GAMS.
- Definir conjuntos de índices de manera compacta. Esta función se explicó en el apartado 10.3.1,
- Indicar errores de compilación en el fichero de salida. Con cuatro asteriscos consecutivos se indica la línea que contiene errores.
- Indicar en el fichero de salida que las restricciones no lineales son infactibles para el punto inicial. En este caso, aparecen tres asteriscos al final de la formulación de la restricción afectada.
- Operador del producto "*" y de la potenciación "**".
- Representar un conjunto comodín cuando se definen vectores, matrices, tablas, variables o restricciones. Este comodín tiene sentido si los índices que recorren dichas estructuras no guardan relación entre sí y, por tanto, no tiene sentido agruparlos en un conjunto. El siguiente ejemplo ayuda a comprender esta característica de GAMS.

```
   Set A conjunto de objetos /a1,a2/;

   Table g(A,*)  características de los objetos
                 alto       ancho      peso
*                (cm)       (cm)       (kg)
       a1        1.0        2.7        3.5
       a2        2.4        1.8        4.4;
```

El código anterior puede reemplazarse por el siguiente:

```
  Sets A conjunto de objetos /a1,a2/
       B conjunto de características /altura, anchura, peso/;

  Table g(A,B)  características de los objetos
                alto       ancho      peso
*               (cm)       (cm)       (kg)
      a1        1.0        2.7        3.5
      a2        2.4        1.8        4.4;
```

10.3.10 Resultados

En el fichero de salida GAMS (fichero con la extensión ".lst") se puede mostrar información adicional a la que aparece por defecto gracias al comando `display`, por ejemplo

```
Display x.l;
```

Para el problema del `transporte`, este comando genera la siguiente información en el fichero de salida GAMS

```
 ----     46 VARIABLE  X.L
cantidad de producto transportada entre i y j en toneladas
            m1          m2          m3
p1     100.000
p2                 200.000     300.000
```

La tabla anterior muestra los valores óptimos del vector x_{ij}. El sufijo `.l` sólo es válido para las variables de optimización. La salida que produce este comando contiene espacios en blanco en las posiciones de los valores por defecto, que en este caso son ceros.

Con el comando `display` se puede conocer el tiempo de cálculo empleado por el optimizador en resolver el problema. Así, la siguiente sentencia GAMS permite conocer el tiempo que transcurre al resolver el problema del `transporte`.

```
Display  transporte.resusd;
```

Esta sentencia sólo es válida después de una sentencia `solve`.

Existe otro comando similar a `display` que permite mostrar los resultados del problema en ficheros de salida alternativos y con el formato que el usuario prefiera. Este comando se explica en el apartado 10.3.14.

10.3.11 Expresiones condicionales

El símbolo"`$`" se puede utilizar en GAMS para manejar subconjuntos cuyos elementos cumplan una determinada condición. La sentencia GAMS

```
demanda(j)$(ord(j) gt 1) ..   sum(i, x(i,j))  =g=  b(j);
```

Tabla 10.8: Ejemplos que muestran la correspondencia entre el operador "$" y la sentencia "if-then-else"

expresión $	expresión if-then-else
`c('p2','m1')$(z.l ge 70)=0.01;`	`If( z.l ge 70,` `c('p2','m1')=0.01;` ` );`
`c('p2','m1')=0.01$(z.l ge 70);`	`If( z.l ge 70,` `c('p2','m1')=0.01;` `else` ` c('p2','m1')=0.0;` ` );`

equivale a

```
demanda2..   sum(i, x(i,'m2'))  =g=  b('m2');
demanda3..   sum(i, x(i,'m3'))  =g=  b('m3');
```

La condición `$(ord(j) gt 1)` establece que existen tantas restricciones de demanda como elementos en el conjunto `j` cuyo ordinal sea mayor que 1. El funcionamiento del operador `ord` se describió en el apartado 10.3.1.

Análogamente, se puede condicionar una suma de variables o la aparición de una variable en una restricción. El siguiente código ilustra estos casos:

```
maximo(i) ..   sum(j$(ord(j) lt card(j)), x(i,j))  =l=
               b(i)$(ord(i) eq 1);
```

y es equivalente a:

```
maximo1..  x('p1','m1') + x('p1','m2') =l= b('p1');
maximo2..  x('p2','m1') + x('p2','m2') =l= 0 ;
```

Las expresiones condicionales típicamente se emplean en la asignación de datos o en las sentencias de escritura de datos de salida (véase el apartado 10.3.14). En ambos casos, el operador "$" puede reemplazarse por una sentencia "if-then-else". En la Tabla 10.8 se muestra un ejemplo de asignación de datos condicionada. El lector debe asegurarse dónde colocar el símbolo '$', pues dependiendo del lugar el resultado de la operación es distinto.

En los apartados 12.1, 12.2.1, y 12.7 se presentan algunos ejemplos adicionales que hacen uso de sentencias condicionales.

En las expresiones condicionales de GAMS se pueden utilizar los operadores típicos de otros lenguajes de programación, como son

- not, and, or, xor como operadores lógicos;
- < (lt), <= (le), = (eq),<> (ne), >= (ge), > (gt) como operadores relacionales.

Tabla 10.9: Ejemplos con conjuntos dinámicos

Expresión matemática	Expresión en GAMS
$k = \{a, b, c\}$	`Set k /a,b,c/`
$s \subseteq k$	`s(k);`
$s = \{a, b, c\}$	s(k)=yes;
$s = \emptyset$	`s(k)=no;`
$s = \{c\}$	`s(k)=no;`
	`s(k)$(ord(k) eq card(k))=yes;`
$s = \{b, c\}$	`s(k)=no;`
	`s(k)$(ord(k) gt 1)=yes;`

10.3.12 Conjuntos dinámicos

Los conjuntos dinámicos son una de las estructuras más útiles que ofrece GAMS. Los elementos de estos conjuntos pueden variar a lo largo de la ejecución del programa. Estos conjuntos siempre se definen como subconjuntos de conjuntos constantes o estáticos. La Tabla 10.9 muestra la equivalencia entre expresiones matemáticas y expresiones GAMS que usan conjuntos dinámicos.

En la primera fila de la Tabla 10.9, se define el conjunto estático `k` y en la segunda su subconjunto dinámico `s(k)`. Para poder utilizar los conjuntos dinámicos es necesario establecer su estado inicial. En la tercera fila se asigna el conjunto vacío como estado inicial. En la cuarta fila se asigna el último elemento del conjunto `k` al subconjunto `s(k)`. Esta asignación hace uso de los operadores `ord(k)` y `card(k)`. El operador '`card`' obtiene el número de elementos de un conjunto estático o dinámico, y el operador `ord` obtiene la posición de un elemento en un conjunto estático (este último operador sólo es válido para conjuntos estáticos). La última fila muestra cómo asignar dos elementos (b y c) al subconjunto dinámico `s(k)`.

Los conjuntos dinámicos también son útiles para operar con conjuntos. Los ejemplos de la Tabla 10.10 muestran las operaciones típicas con conjuntos.

Otros ejemplos donde se emplean conjuntos dinámicos aparecen en el apartado 12.2.1.

10.3.13 Estructuras iterativas

Los bucles, por ejemplo, permiten resolver colecciones de problemas similares que requieren una resolución iterativa debido al empleo de técnicas de descomposición. El código siguiente, basado en el ejemplo 4.1, combina los bucles y los conjuntos dinámicos para resolver un problema iterativamente.

Tabla 10.10: Ejemplos de operaciones con conjuntos

Expresión matemática	Expresión en GAMS
$A = \{a1, a2, a3, a4, a5\}$	`set A conjunto estatico /a1*a5/;`
$b \subseteq A,\ b = \{a1, a2, a5\}$	`set B(A) subconjunto /a1,a2,a5/;`
$c \subseteq A,\ c = \{a2, a3, a5\}$	`set C(A) subconjunto /a2,a3,a5/;`
	`set UN(A) subconjunto dinámico;`
$b \cup c = \{a1, a2, a3, a5\}$	`UN(A)=B(A)+C(A);`
	`set IN(A) subconjunto dinámico;`
$b \cap c = \{a2, a5\}$	`IN(A)=B(A)*C(A);`
	`set COMP(A) subconjunto dinámico;`
$\bar{b} = \{a3, a4\}$	`COMP(A)=not B(A);`
	`set DIFF(A) subconjunto dinámico;`
$b - c = \{a1\}$	`DIFF(A)=B(A)-C(A);`

```
Sets K /1*3/
     DIN(K);

Parameter m(K) pendiente de C2; m('1')=1;
Positive variables x1,x2;
Free variable z; x1.up=3;

Equations
 Cost
 C1
 C2(K)
 C4
 C6;

Cost.. z=e=3*x1+x2; C1.. -x1+x2=l=2; C2(DIN).. x1+m(DIN)*x2=l=6;
           C4.. 2*x1-x2=l=4; C6.. -x1-x2=l=-1;

Model ejcap4 /all/;

loop(K, DIN(K)=yes; Solve ejcap4 using lp maximizing z; m(K+1)=m(K)+1;
);
```

El bucle anterior permite resolver tres problemas similares secuencialmente. La resolución de estos problemas puede ser de utilidad para una análisis de sensibilidad sobre la restricción activa C2 (en cada iteración se añade una restricción que cambia la pendiente de esta recta). En tiempo de ejecución las siguientes restricciones no varían:

```
C1.. - x1 + x2 =l=  2 ;
C4.. 2*x1 - x2 =l=  4 ;
C6.. - x1 - x2 =l= -1 ;
```

Tabla 10.11: Formato de las sentencias `for` y `while`

`scalar contite contador de iteraciones;` `for(contite= 1 to 100,` ` ...` ` ...` `);`	`scalar contite; contite=1;` `while(contite lt 100,` ` ...` ` contite=contite+1;` `);`

Para la primera iteración (`K = 1`), el valor de la función objetivo es 12 (ésta es la solución del problema base) donde la restricción `C2` tiene la forma:

```
C2(1)..   x1 + x2 =l=  6 ;
```

En la segunda iteración (`K = 2`), el valor de la función objetivo es 10 y la restricción `C2` se modela como

```
C2(1)..   x1 + x2 =l=  6 ;
C2(2)..   x1 + 2*x2 =l=  6 ;
```

En la última iteración (`K = 3`), el valor de la función objetivo es 8.9 y la restricción `C2` se toma como

```
C2(1)..   x1 + x2 =l=  6 ;
C2(2)..   x1 + 2*x2 =l=  6 ;
C2(3)..   x1 + 3*x2 =l=  6 ;
```

Nótese que las restricciones `C2` que se han escrito anteriormente no pueden usarse como código en el fichero de entrada GAMS.

En GAMS también se emplean otras estructuras iterativas típicas de los lenguajes de programación, como son `for` y `while`. Estas dos estructuras son de utilidad si el número de iteraciones depende de un escalar y no de un conjunto (véase la Tabla 10.11). La sentencia `while` se usa normalmente cuando el criterio de parada del bucle depende de una expresión condicional. Por otro lado, la sentencia `for` se suele emplear si el número de iteraciones del bucle se conoce con antelación y no depende de ninguna condición.

A continuación se muestra cómo emplear una sentencia `for` en el problema del transporte.

```
...
scalar contite "contador de iteraciones";

for(contite= 1 to 3,
 Solve transporte using lp minimizing z ;
 b(j)=b(j)+ord(j);
);
```

El código equivalente al anterior empleando la sentencia `while` es:

```
...
scalar contite; contite=1;

while(contite le 3,
 Solve transporte using lp minimizing z ;
 b(j)=b(j)+ord(j); contite=contite+1;
);
```

Con cualquiera de los dos códigos se resuelven tres problemas secuencialmente. En la primera iteración (`contite = 1`) GAMS tiene en cuenta el siguiente grupo de restricciones de `demanda`

```
demanda(m1)..  x(p1,m1) + x(p2,m1) =G= 100 ;
demanda(m2)..  x(p1,m2) + x(p2,m2) =G= 200 ;
demanda(m3)..  x(p1,m3) + x(p2,m3) =G= 300 ;
```

Para la segunda iteración (`contite = 2`) el término independiente de las anteriores restricciones se modifica y pasa de ser 100, 200 y 300 a ser 101, 202, y 303, respectivamente. Y, finalmente, en la tercera iteración (`itecount = 3`) los términos independientes son 102, 204, y 306.

10.3.14 Escritura en fichero de salida

El comando `put` permite decidir a los usuarios de GAMS qué ficheros de salida crear, qué información escribir y el formato de escritura. Puede resultar útil para agrupar los resultados en distintos ficheros a conveniencia del usuario. A pesar de la mayor flexibilidad en el formato de escritura de datos, este comando es un poco más complejo de utilizar que el comando `display`. Por ejemplo, para producir la misma salida que el comando `display` en el problema del transporte, se necesitaría el siguiente código:

```
file out /transporte.out/;
put out;
put "----      46 VARIABLE  X.L  cantidad de producto transportada
              entre i y j en toneladas"//;
put "           ";
loop(j,put j.tl:12);
put //;
loop(i,
     put i.tl:2;
     loop(j,
          put$(x.l(i,j) ne 0)  x.l(i,j):12:3;
          put$(x.l(i,j) eq 0) "            ";
     );
     put /;
);
```

En la primera línea del código anterior, se especifica el nombre del fichero de salida mediante el comando `file`. El identificador `out` lo establece el usuario y se refiere al nombre del fichero interno de GAMS que se asocia al fichero de

salida (en este caso, `transporte.out`). A cada fichero de salida debe asignarsele un etiqueta diferente.

Mientras que el comando `display` trata automáticamente los índices de las variables o datos, el comando `put` debe recorrer estos índices mediante estructuras iterativas.

Como se puede observar en el ejemplo anterior, el comando put puede ir seguido de:

- El identificador interno del fichero de salida (véase la segunda línea). Tras esta sentencia, todo se escribirá en el fichero de salida asociado. Si se quiere escribir en un nuevo fichero se debe utilizar un identificador diferente.

- Texto entre comillas. El usuario escribe el texto tal y como quiere que aparezca en el fichero de salida. Ésto se hace en las líneas 3, 4, y 11 del código anterior.

- El símbolo "/". Este símbolo se usa para introducir un retorno de carro en el fichero de salida (véanse las líneas 3, 6, y 13).

- Identificadores de conjuntos, vectores y/o variables seguidos de los sufijos correspondientes. El sufijo `tl` se usa para escribir el identificador de un conjunto, como en la línea 5. En la línea 11 se utiliza el sufijo `l` para escribir el valor de la variable de optimización.

- Una expresión condicional. Si la escritura de los datos depende de que se cumpla una determinada condición, se utiliza el comando "$" (véanse líneas 10 y 11) o la sentencia `if-then-else`.

10.3.15 Listado de las restricciones no lineales

El fichero de salida que genera GAMS automáticamente proporciona cierta información que ayuda al usuario a depurar el programa y detectar errores. No son necesarias muchas nociones de lengua inglesa para entender lo que aparece en este fichero. Sin embargo, cuando se trata de comprobar si las restricciones no lineales están bien formuladas, el fichero de salida no es lo suficientemente autoexplicativo.

El siguiente código representa el problema de minimizar la función $e^{-x_1}x_2^3$, considerando como puntos iniciales $x_1^0 = 1$ y $x_2^0 = 0.5$:

```
 POSITIVE VARIABLE X1,X2; FREE VARIABLE z;
 X1.L=1; X2.L=0.5;

EQUATION
 COSTE    esta es la funcion objetivo;

COSTE..   exp(-X1)*power(X2,3)=e=z;

MODEL nolineal /all/; SOLVE nolineal USING nlp MINIMIZING z;
```

La parte correspondiente a las restricciones no lineales en el fichero de salida es

```
COSTE..- (0.046)*X1+(0.2759)*X2-Z =E= 0; (LHS = 0.046, INFES = 0.046 ***)
```

En la expresión anterior, los coeficientes entre paréntesis representan la evaluación de la primera derivada de la restricción `COSTE` respecto a cada variable en el punto inicial; es decir, $-e^{-x_1^0} \cdot (x_2^0)^3 = -0.046$ y $e^{-x_1^0} \cdot 3 \cdot (x_2^0)^2 = 0.2759$. Además, se observan dos igualdades después del punto y coma. La primera igualdad es la evaluación del término dependiente (LHS) de la restricción `COSTE` para el punto inicial: $e^{-x_1^0} \cdot (x_2^0)^3 = 0.046$. La segunda indica que esta restricción es infactible en el punto inicial, aunque no significa que sea infactible en cualquier punto.

Capítulo 11

Algunos ejemplos en GAMS

11.1 Introducción

Este capítulo tiene como finalidad mostrar al lector cómo formular y resolver en GAMS los ejemplos planteados en capítulos anteriores. Para el primer ejemplo se explican en detalle los comandos de GAMS necesarios para su formulación, sin embargo, en los demás, con objeto de evitar explicaciones repetitivas, sólo se comentan las novedades.

Para mayor claridad, las palabras reservadas de GAMS y los identificadores correspondientes a los datos de entrada se escriben en mayúscula y el resto del código se escribe en minúsculas. Para evitar problemas a los lectores, no se incluyen acentos en el código GAMS, ya que GAMS no admite acentos, a menos que se utilicen comandos especiales.

11.2 Ejemplos de programación lineal

Este apartado muestra el código GAMS para los ejemplos de programación lineal presentados en capítulos anteriores.

11.2.1 El problema del transporte

El problema del transporte se planteó en el apartado 1.2, y en el ejemplo 1.1. Este ejemplo se formula a continuación. Minimizar

$$Z = \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} \tag{11.1}$$

sujeto a

$$
\begin{array}{rcll}
\sum\limits_{j=1}^{n} x_{ij} & = & u_i; & \forall i = 1, \dots, m \\
\sum\limits_{i=1}^{m} x_{ij} & = & v_j; & \forall j = 1, \dots, n \\
x_{ij} & \geq & 0; & \forall i = 1 \dots m; \ \forall j = 1, \dots, n
\end{array}
\tag{11.2}
$$

donde $m = n = 3$ y

$$
\mathbf{C} = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 1 & 2 \\ 3 & 2 & 1 \end{pmatrix}, \quad \mathbf{u} = \begin{pmatrix} 2 \\ 3 \\ 4 \end{pmatrix}, \text{ y } \mathbf{v} = \begin{pmatrix} 5 \\ 2 \\ 2 \end{pmatrix}
$$

Desde el punto de vista de un usuario GAMS, se deben tener en cuenta las siguientes consideraciones:

- Este problema tiene tres tipos de restricciones: (1) una restricción que representa la función objetivo, (2) un grupo de restricciones formado por tantas restricciones como orígenes, y (3) un grupo de restricciones formado por tantas restricciones como destinos. Por tanto, se deben definir dos conjuntos de índices, uno para referirse a los orígenes y otro para los destinos, de forma que las restricciones del tipo (2) y (3) se puedan escribir de forma compacta.

- Se tienen una matriz y dos vectores de datos: c_{ij}, u_i, v_j. Los vectores u_i y v_j se han de declarar mediante el comando `PARAMETER`, puesto que son unidimensionales. Sin embargo, para declarar la matriz c_{ij} se pueden utilizar los comandos `PARAMETER` o `TABLE`. En este caso se ha empleado el comando `TABLE`.

- Para almacenar el valor de la función objetivo es necesario declarar una variable no acotada (`z` en este caso).

- El vector x_{ij} es el vector de variables a optimizar. Obsérvese que existen tantas variables como el resultado de multiplicar el número de orígenes por el número de destinos.

- Una vez que se han declarado las variables, es conveniente establecer su naturaleza. Así, las variables x_{ij} son positivas, y la variable `z` no está restringida en signo (por defecto `FREE`).

- Después de declarar los conjuntos, los datos y las variables de optimización, se deben declarar y definir las restricciones del problema. La función objetivo se declara como una sola restricción (no depende de ningún conjunto de índices). Por otro lado, se deben declarar los dos grupos de restricciones. El primer grupo, `ENVIA(I)`, equivale a tantas restricciones como orígenes (por eso depende del conjunto I de orígenes). El segundo grupo, `RECIBE(J)`, equivale a tantas restricciones como destinos (por eso depende del conjunto J de destinos).

- Una vez que se han declarado las restricciones, éstas deben definirse. Esta tarea resulta sencilla dada la similitud entre la formulación matemática y la formulación en GAMS. Obsérvese que en GAMS el término $\sum_{j=1}^{n} x_{ij}$ se escribe como `SUM(J,x(I,J))`.

- A continuación es necesario identificar el modelo a resolver (`transport`) y establecer cuáles son las restricciones que lo componen. En este caso, el modelo está formado por todas las restricciones, es decir, `COSTE`, `ENVIA(I)`, y `RECIBE(J)`. Cuando se define el modelo sólo se especifica la etiqueta con la que se declararon las restricciones y no los conjuntos de los que éstas puedan depender. Puesto que en este modelo se incluyen todas las restricciones definidas, se puede escribir la palabra reservada `ALL` en lugar de listar una a una las restricciones.

- Con el modelo definido, el siguiente paso es ordenar su resolución. Para indicar que se trata de un problema de programación lineal de minimización, en la orden de resolución se requieren las palabras reservadas `LP` y `MINIMIZING`. También se indica la variable que representa la función objetivo a minimizar.

Un posible fichero de entrada donde se formula y resuelve el ejemplo del transporte es el siguiente.

```
$title EL PROBLEMA DEL TRANSPORTE

**  Primero se declaran y definen los conjuntos de indices.
**  El conjunto I  recorre los origenes.
**  El conjunto I recorre los destinos.
**  El simbolo ‘*’ permite compactar el listado de indices.

SETS
 I conjunto de indices para recorrer los origenes   /I1*I3/
 J conjunto de indices para recorrer los destinos   /J1*J3/;

**  Los vectores de datos (U(I) y V(J)) se declaran como parametros
**  a los que se les asignan los datos de entrada.

PARAMETERS
 U(I)  cantidad a transportar desde el origen I
       /I1 2
        I2 3
        I3 4/
 V(J)  cantidad recibida en el destino J
       /J1 5
        J2 2
        J3 2/;

**  La matriz de datos C(I,J) se define como una tabla.

TABLE C(I,J) coste de transporte entre el origen I y el destino J
      J1    J2   J3
I1     1     2    3
I2     2     1    2
I3     3     2    1;
```

```
**  A continuacion se declaran las variables de optimizacion.
**  Primero se declara la funcion objetivo.
**  Despues se declara el resto de variables.

VARIABLES
 z      valor de la funcion objetivo
 x(I,J) la cantidad de producto a transportar desde I a J;

**  En la siguiente linea de codigo se establece la naturaleza de las variables.
**  En este problema todas las variables son positivas excepto z.

POSITIVE VARIABLE x(I,J);

**  Se declara en primer lugar la funcion objetivo.
**  El resto de restricciones se declaran utilizando
**  la notacion compacta de GAMS con conjuntos de indices.

EQUATIONS
 COSTE      funcion objetivo
 ENVIA(I)   ecuacion de envios
 RECIBE(J)  ecuacion de recepcion;

**  Las siguientes sentencias formulan las restricciones anteriores.
**  Todas las restricciones son de igualdad (=E=).
**  La funcion objetivo es un sumatorio en I y en J de las variables x(I,J).
**  La segunda restriccion equivale a tres ecuaciones (conjunto I).
**  El termino de la izquierda es una suma en J de las variables x(I,J), y
**  el termino de la derecha es el vector de datos U(I).
**  Analogamente se formulan las restricciones ENVIA.

COSTE..        z=E=SUM((I,J),C(I,J)*x(I,J));
ENVIA(I) ..    SUM(J,x(I,J))=E=U(I);
RECIBE(J) ..   SUM(I,x(I,J))=E=V(J);

**  La siguiente sentencia identifica el modelo.

MODEL transporte /COSTE,ENVIA,RECIBE/;

**  La siguiente sentencia hace que GAMS resuelva el modelo
**  mediante un optimizador de programacion lineal.

SOLVE transporte USING lp MINIMIZING z;
```

A continuación se muestra parte del fichero de salida.

```
                       LOWER     LEVEL     UPPER    MARGINAL
---- VAR Z              -INF     14.000     +INF       .
  Z           valor de la funcion objetivo

---- VAR X           la cantidad de producto a transportar desde I a J
         LOWER     LEVEL     UPPER    MARGINAL
I1.J1      .        2.000     +INF       .
I1.J2      .         .        +INF      2.000
I1.J3      .         .        +INF      4.000
I2.J1      .        1.000     +INF       .
I2.J2      .        2.000     +INF       .
```

```
I2.J3      .        .       +INF     2.000
I3.J1      .      2.000     +INF       .
I3.J2      .        .       +INF      EPS
I3.J3      .      2.000     +INF       .
```

En este fichero de salida, se muestra el valor óptimo de las variables en la columna `LEVEL`. Por tanto, la solución óptima del problema del transporte es

$$Z = 14, \qquad \mathbf{X} = \begin{pmatrix} 2 & 0 & 0 \\ 1 & 2 & 0 \\ 2 & 0 & 2 \end{pmatrix}$$

11.2.2 El problema de planificación de la producción

El problema de planificación de la producción se estudió en detalle en el apartado 1.3, y en el ejemplo 1.2. A continuación, se resuelve un ejemplo de este problema. Maximizar

$$Z = \sum_{t=1}^{n} (a_t y_t - b_t x_t - c_t s_t) \tag{11.3}$$

sujeto a

$$\begin{array}{lcl} s_{t-1} + x_t - s_t & = & y_t, \ \forall t = 1, \ldots, n \\ s_t, x_t, y_t & \geq & 0 \end{array} \tag{11.4}$$

donde $a_t = b_t = c_t = 1$, $\forall t = 1 \ldots n$; $n = 4$; $s_0 = 2$ y $\mathbf{y} = (2, 3, 6, 1)^T$.

Los comentarios sobre el código GAMS anterior son:

- Se emplea el sufijo `fx` para fijar el valor de un elemento dentro de un vector de variables.
- En la restricción correspondiente a la función objetivo, se emplea el comando `$` para sumar algunos elementos (no todos) que forman parte de un vector. Asimismo, este comando condicional se emplea para definir un grupo de restricciones que sólo tienen sentido para un subconjunto de índices.

El fichero de entrada para este problema es

```
$title PROBLEMA DE LA PLANIFICACION DE LA PRODUCCION

**  Se declara el conjunto de indices T.
**  El conjunto tiene 5 elementos que se definen
**  mediante el simbolo '*'.

SET
 T   Indice de los meses /0*4/;

**  Una vez se define el conjunto T, se declara el vector de datos Y(T)
**  y se asignan valores a todos elementos excepto al primero
**  (elemento '0') que, por defecto, toma el valor cero.
```

```
PARAMETER

 Y(T) demanda en el mes T
   /1 2
    2 3
    3 6
    4 1/
 A(T)
 B(T)
 C(T);

** Se asigna el valor 1 a todos los elementos de los vectores A, B y C
** (se emplea el conjunto T).

A(T)=1; B(T)=1; C(T)=1;

**  Se declaran las variables de optimizacion.
**  Primero se declara el valor de la funcion objetivo (z).
**  Esta variable no depende de ningun conjunto
**  puesto que es una variable de tipo escalar.
**  El resto de variables depende del conjunto T, por lo que
**  se declaran haciendo referencia a este conjunto.

VARIABLES

 z    valor de la funcion objetivo
 x(T) numero de unidades producidas en el mes T
 s(T) numero de unidades almacenadas en el mes T;

**  A excepcion del valor de la funcion objetivo,
**  que, por defecto, no esta restringido en signo, las demas
**  variables del problema se definen como positivas.

POSITIVE VARIABLES x(T),s(T);

**  La siguiente orden asigna un valor inicial a la variable s(T).
**  En el mes '0', esta variable toma el valor 2.

s.fx('0')=2;

**  Se declara la funcion objetivo, y las 5 restricciones, ecuaciones INOUT.


EQUATIONS
 COST     funcion objetivo
 INOUT(T) balance de entradas y salidas;

**  Una vez declaradas las restricciones se pasa a su definicion.
**  La primera restriccion es la funcion objetivo.
**  Esta ecuacion presenta una suma condicionada ($).
**  Solo se consideran los 4 ultimos elementos del conjunto T
**  al emplear la expresion '$(ord(T) gt 1)'.
**  A continuacion se definen las 4 ecuaciones INOUT.
**  Estas ecuaciones se declararon como dependientes del conjunto T.
**  Dado que se hace referencia al elemento T-1 en el termino s(T-1) de la
**  restriccion, se debe partir del segundo elemento del conjunto T.
```

```
**  Para asegurar que las ecuaciones INOUT no representan el 'mes 0',
**  se emplea la expresion condicional `$(ord(T) gt 1)' despues
**  del nombre de la ecuacion.

COST..z=E= SUM(T$(ord(T) gt 1), A(T)*Y(T)-B(T)*x(T)-C(T)*s(T));
INOUT(T)$(ord(T) gt 1)..s(T)=E= s(T-1)+x(T)-Y(T);

**  El comando siguiente declara y define el modelo,
**  en el que se incluyen todas las restricciones definidas.

MODEL planificacion /ALL/;

**  El comando siguiente solicita que GAMS resuelva el problema de la
**  planificacion mediante un optimizador de programacion lineal
**  que maximice el valor de la funcion objetivo z.

SOLVE planificacion USING lp MAXIMIZING z;
```

A continuación, se muestra un extracto del fichero de salida GAMS:

```
                       LOWER     LEVEL     UPPER    MARGINAL
---- VAR Z              -INF      2.000     +INF       .
  Z           valor de la funcion objetivo

---- VAR X           numero de unidades producidas en el mes T
     LOWER     LEVEL     UPPER    MARGINAL
1      .         .        +INF       .
2      .        3.000     +INF       .
3      .        6.000     +INF       .
4      .        1.000     +INF       .

---- VAR S           numero de unidades almacenadas en el mes T
     LOWER     LEVEL     UPPER    MARGINAL
0     2.000     2.000     2.000     1.000
1      .         .        +INF     -1.000
2      .         .        +INF     -1.000
3      .         .        +INF     -1.000
4      .         .        +INF     -2.000
```

Por tanto, la solución del problema de planificación de la producción es

$$Z = 2, \qquad \mathbf{x} = (0,3,6,1)^T, \qquad \mathbf{s} = (2,0,0,0,0)^T$$

11.2.3 El problema de la dieta

Este problema se estudió en el apartado 1.4, y en el ejemplo 1.3. A continuación se resuelve un ejemplo de este problema. Minimizar

$$Z = \sum_{j=1}^{n} c_j x_j \tag{11.5}$$

sujeto a

$$\begin{array}{rcl} \sum_{j=1}^{n} a_{ij}x_j & \geq & b_i; \quad i = 1, \dots, m \\ x_j & \geq & 0; \quad j = 1, \dots, n. \end{array}$$

donde $m = 4$, $n = 5$ y

$$\mathbf{A} = \begin{pmatrix} 78.6 & 70.1 & 80.1 & 67.2 & 77.0 \\ 6.50 & 9.40 & 8.80 & 13.7 & 30.4 \\ 0.02 & 0.09 & 0.03 & 0.14 & 0.41 \\ 0.27 & 0.34 & 0.30 & 1.29 & 0.86 \end{pmatrix}, \quad \mathbf{b} = \begin{pmatrix} 74.2 \\ 14.7 \\ 0.14 \\ 0.55 \end{pmatrix}, \quad \mathbf{c} = \begin{pmatrix} 1 \\ 0.5 \\ 2 \\ 1.2 \\ 3 \end{pmatrix}$$

Este ejemplo no requiere el uso de nuevas expresiones en GAMS. Para comprender el siguiente código basta con leer los comentarios que tiene insertados:

```
$title PROBLEMA DE LA DIETA

**  Se definen los conjuntos de indices:
**  El indice I se emplea para hacer referencia a los 4 nutrientes.
**  El indice J se emplea para hacer referencia a los 5 alimentos.

SET
 I    conjunto de nutrientes   /DN,DP,Ca,Ph/
 J    conjunto de alimentos    /MaizA,Avena,MaizB,Salvado,Linaza/;

**  Los vectores para almacenar los datos de entrada se declaran como parametros.
**  Se asignan los datos a estos vectores.

PARAMETERS
 B(I) cantidad minima que se necesita del nutriente I
      /DN 74.2
       DP 14.7
       Ca 0.14
       Ph 0.55/
 C(J)  coste de una unidad de alimento J
       /MaizA    1
        Avena    0.5
        MaizB    2
        Salvado  1.2
        Linaza   3/;

**  Se declara y define la matriz de datos como una tabla

TABLE A(I,J) la cantidad de nutriente I en una unidad de alimento J
     MaizA Avena MaizB SalvadoLinaza
DN   78.6  70.1  80.1  67.2   77.0
DP    6.5   9.4   8.8  13.7   30.4
Ca   0.02  0.09  0.03  0.14   0.41
Ph   0.27  0.34  0.30  1.29   0.86;

**  A continuacion se declaran las variables de optimizacion.
**  Primero se declara el valor de la funcion objetivo.
**  Despues se declara el resto de variables indicando su dimension.

VARIABLES
  z    valor de la funcion objetivo
  x(J) cantidad a comprar del alimento J;

**  El comando siguiente establece la naturaleza de las variables.
**  En el problema de la dieta, todas las variables son positivas
```

```
**  excepto la que representa el valor de la funcion objetivo.

POSITIVE VARIABLE x(J);

**  Se declaran las restricciones del problema.
**  En primer lugar, la restriccion
**  correspondiente a la funcion objetivo (COST).
**  Despues, las 4 restricciones restantes (NUTFOOD), que
**  se declaran dependiendo del conjunto I.

EQUATIONS
 COST       funcion objetivo
 NUTFOOD(I) relacion entre nutrientes y alimentos;

**  La funcion objetivo es una restriccion de igualdad (=E=).
**  Las restricciones NUTFOOD son restricciones de desigualdad (=G=).

COST ..         z=E=SUM(J,C(J)*x(J));
NUTFOOD(I) ..   SUM(J,A(I,J)*x(J))=G=B(I);

**  Los comandos siguientes declaran y definen el modelo dieta,
**  incluyendo todas las restricciones declaradas, y solicitan
**  a GAMS que resuelva el problema mediante un optimizador lineal.

MODEL dieta /ALL/;
SOLVE dieta USING lp MINIMIZING z;
```

Un extracto del fichero de salida GAMS, que muestra la solución óptima del problema de la dieta, es el siguiente:

```
                       LOWER     LEVEL     UPPER    MARGINAL
---- VAR Z              -INF      0.793     +INF       .
  Z           valor de la funcion objetivo

---- VAR X           cantidad a comprar del alimento J
           LOWER     LEVEL     UPPER    MARGINAL
MaizA        .         .        +INF      0.634
Avena        .       1.530      +INF       .
MaizB        .         .        +INF      1.543
Salvado      .       0.023      +INF       .
Linaza       .         .        +INF      1.525
```

Es decir, que la solución óptima del problema de la dieta es

$$Z = 0.793, \qquad \mathbf{x} = (0, 1.53, 0, 0.023, 0)^T \tag{11.6}$$

11.2.4 El problema de flujos en redes

El problema de los flujos en redes, que se estudió en el apartado 1.5, y en el ejemplo 1.4, se puede formular como sigue.

Minimizar

$$Z = \sum_{ij} c_{ij} x_{ij}$$

sujeto a

$$\begin{array}{rcll} \sum_j (x_{ij} - x_{ji}) & = & f_i; & i = 1, \dots, n \\ -f_{ij} \leq x_{ij} & \leq & f_{ij}; & \forall i < j \end{array}$$

donde $n = 4$; se supone que $f_{ij} = 4, \forall i, j$, $(f_1, f_2, f_3, f_4) = (7, -4, -1, -2)$ y $c_{ij} = 1; \forall i, j$.

En la implementación de este ejemplo se ha explotado la versatilidad de los subconjuntos en GAMS mediante:

- La definición de un subconjunto para establecer las conexiones entre los nudos
- La inclusión de subconjuntos en la definición de ecuaciones para que en éstas sólo se tenga en cuenta los nudos que están conectados

Un fichero de entrada en GAMS para este problema es

```
$title PROBLEMA DE FLUJO EN REDES

**  En primer lugar se declaran los conjuntos:
**  El conjunto I tiene 4 elementos.
**  El subconjunto CONEX depende del conjunto I.
**  El subconjunto CONEX define las conexiones validas entre los nudos I.

SET
 I          conjunto de nudos en la red  /I1*I4/
 CONEX(I,I) conjunto de conexiones de nudos  /I1.I2,I1.I3,I1.I4,I2.I4,I3.I4/;

**  El conjunto de nudos I se duplica para hacer referencias
**  a distintos elementos del mismo dentro de la misma restriccion.

 ALIAS(I,J)

**  Los vectores para almacenar los datos de entrada se declaran como parametros.
**  Se asignan los datos a estos vectores.
**  FMAX(I,J) es una matriz de datos, declarada como un parametro,
**  cuyos elementos toman el valor 4. Esta es una forma compacta
**  de declarar una matriz (no es necesario usar el comando TABLE).

PARAMETERS
 F(I)   flujo de entrada y salida en el nudo I
      /I1   7
       I2  -4
       I3  -1
       I4  -2/
 FMAX(I,J) capacidad maxima de la conexion entre I y J;

FMAX(I,J)=4;

**  Se declaran las variables de optimizacion.

VARIABLES
  z      valor de la funcion objetivo
  x(I,J) flujo que sale desde el nudo I hacia J;
```

```
**  El limite superior de las variables de optmizacion es
**  la capacidad maxima de las conexiones.

x.lo(I,J)=-FMAX(I,J);
x.up(I,J)=FMAX(I,J);

** Se declaran las restricciones.

EQUATIONS
 COST       funcion objetivo
 BALANCE(I)  condicion de conservacion del flujo;

**  La funcion objetivo solo considera los nudos conectados.
**  Esto se consigue mediante la expresion condicional $CONEX(I,J).
**  Las 4 ecuaciones de BALANCE solo consideran el flujo entre
**  nudos conectados y por eso tambien se incluye $CONEX(I,J).

COST ..         z   =E=  SUM(CONEX(I,J),x(I,J)) ;
BALANCE(I) ..   SUM(J$CONEX(I,J),x(I,J))-SUM(J$CONEX(J,I),x(J,I))  =E=  F(I) ;

**  Los comandos siguientes declaran y definen el modelo flujored,
**  incluyendo todas las restricciones declaradas, y solicitan
**  a GAMS que resuelva el problema mediante un optimizador lineal.

MODEL flujored /ALL/;
SOLVE flujored USING lp MINIMIZING z;
```

La solución del problema en formato GAMS es:

```
                       LOWER     LEVEL     UPPER    MARGINAL
---- VAR Z              -INF      5.000     +INF       .
  Z          valor de la funcion objetivo

---- VAR X           flujo que sale desde el nudo I hacia J
         LOWER     LEVEL     UPPER    MARGINAL
I1.I2    -4.000      .        4.000      .
I1.I3    -4.000     3.000     4.000      .
I1.I4    -4.000     4.000     4.000    -1.000
I2.I4    -4.000    -4.000     4.000     EPS
I3.I4    -4.000     2.000     4.000      .
```

Por tanto, la solución del problema es

$$Z = 5, \;\; x_{12} = 0, \;\; x_{13} = 3, \;\; x_{14} = 4, \;\; x_{24} = -4, \;\; x_{34} = 2$$

El problema de una red de abastecimiento de agua

Este problema se estudió en el apartado 1.5, y se formula como sigue.

Minimizar

$$Z = \sum_{ij} |x_{ij}|$$

sujeto a

$$\begin{array}{rcll} \sum_j (x_{ij} - x_{ji}) & = & f_i; & i = 1, \ldots, n \\ -f_{ij} \;\; \leq \;\; x_{ij} & \leq & f_{ij}; & \forall i < j \end{array}$$

donde se supone que $f_{ij} = 8, \forall i, j$, y $(f_1, f_2, f_3, f_4) = (20, -3, -10, -7)$.

Algunas consideraciones destacables para este ejemplo son:

- Se puede evitar el empleo de la función valor absoluto (función no lineal) si se considera que una variable no restringida en signo es la diferencia de dos variables positivas. Por tanto, la versión lineal del problema anterior es como sigue.

  Minimizar
  $$Z = \sum_{ij} (x_{ij}^+ + x_{ij}^-)$$
  sujeto a
  $$\begin{array}{rcl} \sum_j \left[(x_{ij}^+ - x_{ij}^-) - (x_{ji}^+ - x_{ji}^-) \right] & = & f_i; \quad i = 1, \ldots, n \\ -f_{ij} \leq x_{ij}^+ - x_{ij}^- & \leq & f_{ij}; \quad \forall i < j \end{array}$$
  donde $x_{ij}^+ > 0, \forall i, j$, y $x_{ij}^- > 0, \forall i, j$

- Mediante el uso de un subconjunto en las restricciones `UPLIMIT` y `LOLIMIT` sólo se consideran las conexiones entre nudos que dicta el subconjunto.

```
$title RED DE ABASTECIMIENTO DE AGUA (lineal)

**  En primer lugar se declaran los conjuntos:
**  El conjunto I tiene 4 elementos.
**  CONEX se define como un subconjunto de I.
**  El subconjunto CONEX define las conexiones validas entre los nudos I.

SET
 I         conjunto de nudos de la red /I1*I4/
 CONEX(I,I) conjunto de conexiones entre nudos /I1.I2,I1.I3,I1.I4,I2.I4,I3.I4/;

**  El conjunto de nudos I se duplica para hacer referencias
**  a distintos elementos del mismo dentro de la misma restriccion.

 ALIAS(I,J)

**  Los vectores para almacenar los datos de entrada se declaran como parametros.
**  Se asignan los datos a estos vectores.
**  FMAX(I,J) es una matriz de datos, declarada como un parametro,
**  cuyos elementos toman el valor 8. Esta es una forma compacta
**  de declarar una matriz (no es necesario usar el comando TABLE).

PARAMETERS
 X(I,J) flujo total que parte del nudo I hacia el nudo J
 F(I)   flujo de entrada y salida en el nudo I
      /I1  20
       I2  -3
       I3  -10
       I4  -7/
 FMAX(I,J) capacidad maxima de la conexion entre I y J;

FMAX(I,J)=8;
```

```
**  Se declaran las variables de optimizacion.

VARIABLES
  z       valor de la funcion objetivo
  xp(I,J) parte positiva de x
  xn(I,J) parte negativa de x;

**  El flujo entre I y J se representa mediante variables positivas.

POSITIVE VARIABLES xp(I,J),xn(I,J);

** Se declaran las restricciones

EQUATIONS
 COST          funcion objetivo
 BALANCE(I)    condicion de conservacion del flujo
 UPLIMIT(I,J)  limite superior del flujo entre el nudo I y J
 LOLIMIT(I,J)  limite superior del flujo entre el nudo I y J;

**  La funcion objetivo es la suma de los valores absolutos de los flujos
**  entre los nudos. Para que solo se sumen los nudos conectados
**  se utiliza la condicion $CONEX(I,J).
**  Las 4 ecuaciones de BALANCE solo consideran el flujo entre
**  nudos conectados y por eso tambien se incluye $CONEX(I,J).
**  El flujo entre I y J esta limitado por la capacidad maxima
**  de las conexiones mediante las restricciones UPLIMIT(I,J) y
**  LOLIMIT(I,J).


COST ..          z   =E=  SUM((I,J)$CONEX(I,J),xp(I,J)-xn(I,J)) ;
BALANCE(I) ..    SUM(J$CONEX(I,J),xp(I,J))-SUM(J$CONEX(J,I),xp(J,I))-
                 SUM(J$CONEX(I,J),xn(I,J))+SUM(J$CONEX(J,I),xn(J,I))  =E=
F(I) ;
UPLIMIT(I,J)$(CONEX(I,J)) .. xp(I,J) - xn(I,J) =l=  FMAX(I,J);
LOLIMIT(I,J)$(CONEX(I,J)) .. xp(I,J) - xn(I,J) =g= -FMAX(I,J);

**  Los comandos siguientes declaran y definen el modelo,
**  incluyendo todas las restricciones declaradas, y solicitan
**  a GAMS que resuelva el problema mediante un optimizador lineal.

MODEL raa /ALL/;
SOLVE raa USING lp MINIMIZING z;

**  El comando siguiente asigna a x(I,J) la diferencia entre
**  su parte positiva y negativa. Esto es necesario porque
**  se ha empleado la version lineal de la funcion
**  valor absoluto.

x(I,J)=xp.L(I,J)-xn.L(I,J);

**  El comando siguiente permite imprimir el valor de
**  x(I,J) en el fichero de salida (*.lst).

 DISPLAY x;
```

Parte del fichero de salida GAMS es:

```
                       LOWER     LEVEL     UPPER    MARGINAL
```

```
---- VAR Z              -INF     23.000     +INF       .
  Z           valor de la funcion objetivo

---- VAR XP          parte positiva de x
        LOWER     LEVEL     UPPER    MARGINAL
I1.I2      .        4.000     +INF       .
I1.I3      .        8.000     +INF       .
I1.I4      .        8.000     +INF       .
I2.I4      .        1.000     +INF       .
I3.I4      .         .        +INF      2.000

---- VAR XN          parte negativa de x
        LOWER     LEVEL     UPPER    MARGINAL
I1.I2      .         .        +INF      2.000
I1.I3      .         .        +INF      2.000
I1.I4      .         .        +INF      2.000
I2.I4      .         .        +INF      2.000
I3.I4      .        2.000     +INF       .

----      84 PARAMETER X             flujo total que parte del nudo I hacia el nudo J
           I2          I3          I4
I1       4.000       8.000       8.000
I2                               1.000
I3                               2.000
```

La solución del problema es:

$$Z = 23, \quad x_{12} = 4, \quad x_{13} = 8, \quad x_{14} = 8, \quad x_{24} = 1, \quad x_{34} = 2$$

11.2.5 El problema de la cartera de valores

Este problema se estudió en el apartado 1.6, y se puede formular como sigue.

Maximizar

$$Z = \sum_j d_j(b_j + x_j) \tag{11.7}$$

sujeto a

$$\begin{array}{rcl} b_i + x_i & \geq & 0 \\ r(\sum_j v_j(b_j + x_j)) & \leq & v_i(b_i + x_i) \\ \sum_j v_j x_j & = & 0 \\ \sum_j w_j(b_j + x_j) & \geq & (1+s)\sum_j v_j b_j \end{array} \tag{11.8}$$

Considérese el caso en el que se tiene un conjunto de acciones de 3 valores distintos, 75 de A, 100 de B, y 35 de C, cuyos precios son 20 dólares, 20 dólares, y 100 dólares, respectivamente. Además, se sabe que: A no genera dividendo y su precio futuro es 18 dólares, B genera 3 dólares por acción y su precio futuro es 23 dólares, y C genera 5 dólares por acción y su precio futuro es 102 dólares. Si se toman los porcentajes r, como 0.25 y s, 0.03, las restricciones anteriores

se escriben como

$$
\begin{array}{rcl}
x_A & \geq & -75 \\
x_B & \geq & -100 \\
x_C & \geq & -35 \\
0.25\,[20(75+x_A)+20(100+x_B)+100(35+x_C)] & \leq & 20(75+x_A) \\
0.25\,[20(75+x_A)+20(100+x_B)+100(35+x_C)] & \leq & 20(100+x_B) \\
0.25\,[20(75+x_A)+20(100+x_B)+100(35+x_C)] & \leq & 100(35+x_C) \\
20x_A+20x_B+100x_C & = & 0 \\
18(75+x_A)+23(100+x_B)+102(35+x_C) & \geq & 1.03(20(175)+3500)
\end{array}
\tag{11.9}
$$

The código GAMS que permite la resolución de este ejemplo es:

```
$title EL PROBLEMA DE LA CARTERA DE VALORES

**  Se declara el conjunto I con 3 elementos.

SET
 I         conjunto de valores  /A,B,C/;

**  El conjunto de acciones I se duplica para hacer referencias
**  a distintos elementos del mismo dentro de la misma restriccion.

ALIAS(I,J);

SCALARS  r  porcentaje       /0.25/
         s  porcentaje       /0.03/;

TABLE  data(I,*)
        B     V    D    W
*            ($)  ($)  ($)
 A      75   20    0   18
 B     100   20    3   23
 C      35  100    5  102;

VARIABLES
  z    valor de la funcion objetivo
  x(I) numero de acciones del valor I;
POSITIVE VARIABLE x(I);

x.lo(I)=-data(I,'B');

EQUATIONS
 COST        funcion objetivo
 NOCHANGE    no cambia el valor actual
 INFLATION   el capital futuro debe ser un 3% mayor que el capital invertido
actualmente
 BALANCE(I)  se evita demasiada confianza en un valor concreto;

COST ..        z=E=SUM(I,data(I,'D')*(x(I)+data(I,'B')));
NOCHANGE ..    SUM(I,data(I,'V')*x(I))=E=0;
INFLATION ..   SUM(I,data(I,'W')*(x(I)+data(I,'B')))=G=
               (1+s)*SUM(I,data(I,'V')*data(I,'B'));
BALANCE(I)..   r*SUM(J,data(J,'V')*(x(J)+data(J,'B')))=L=
               data(I,'V')*(x(I)+data(I,'B'));
```

```
MODEL cartera /ALL/;
SOLVE cartera USING lp MAXIMIZING z;
```

Parte del fichero de la salida GAMS es:

```
                      LOWER     LEVEL     UPPER    MARGINAL
---- VAR Z             -INF    612.500     +INF       .
  Z           valor de la funcion objetivo

---- VAR X          numero de acciones del valor I
     LOWER     LEVEL     UPPER    MARGINAL

A   -75.000    12.500     +INF       .
B  -100.000    75.000     +INF       .
C   -35.000   -17.500     +INF       .
```

La solución para este ejemplo es:

$Z = 612.5$ que se alcanza para los valores $x_A = 12.5, \;\; x_B = 75.0, \;\; x_C = -17.5$

11.2.6 El sistema de vigas y cuerdas

Este ejemplo se estudió en el apartado 1.7. Un caso particular es el siguiente modelo. Maximizar

$$Z = \sum_i x_i$$

sujeto a

$$\begin{array}{rcl}
T_E + T_F &=& x_2 \\
T_C + T_D &=& T_F \\
T_A + T_B &=& x_1 + T_C + T_D \\
10T_F &=& 5x_2 \\
8T_D &=& 6T_F \\
10T_B &=& 5x_1 + 2T_C + 10T_D
\end{array}$$

El lector puede observar lo siguiente en el código GAMS correspondiente a este problema:

- La formulación en GAMS es válida para cualquier topología de vigas y cuerdas.
- Se necesitan subconjuntos para escribir las restricciones en notación compacta.
- Para establecer las direcciones de las cargas se les asocia un signo.

El fichero de entrada en GAMS que resuelve la versión lineal de este problema es

```
$title EL PROBLEMA DE LAS VIGAS Y LAS CUERDAS (LINEAL)

**  Se definen los indices en primer lugar:
**  Indice B para referirse a las 3 vigas.
```

```
**  Indice R para referirse a las 6 cuerdas.
**  Indice L para referirse a las 2 cargas.

SET
 B    conjunto de vigas    /B1*B3/
 R    conjunto de cuerdas  /RA,RB,RC,RD,RE,RF/
 L    conjunto de cargas   /L1,L2/

UPP(B,R)
      / B1.(RA,RB)
        B2.(RC,RD)
        B3.(RE,RF)
      /
DOWN(B,R)
      / B1.(RC,RD)
        B2.(RF)
      /
LOAD(B,L)
      / B1.L1
        B3.L2
      /;

**  Los vectores para almacenar los datos de entrada se declaran como parametros.
**  Se asignan los datos a estos vectores.

PARAMETER LMAX(R) carga maxima permitida en la cuerda R
  / (RA,RB) 300
    (RC,RD) 200
    (RE,RF) 100
  /;
PARAMETER DL(L) coordenadas de la carga L
  / L1  7
    L2  5
  /;

PARAMETER DR(R) coordenadas de la cuerda R
  / RA  2
    RB 12
    RC  4
    RD 12
    RE  0
    RF 10 /;

VARIABLES
  z    valor de la funcion objetivo
  x(L) carga aplicada
  t(R) tension en la cuerda R ;

t.up(R) = LMAX(R);

** Se declaran las restricciones del problema.

EQUATIONS
 COST       funcion objetivo
 FORCES(B)  condicion de equilibrio de fuerzas
 MOMENT(B)  condicion de equilibrio de momentos;
```

```
COST ..         z  =E=  SUM(L, x(L)) ;

FORCES(B)..     SUM(R$UPP(B,R),t(R))=E= SUM(L$LOAD(B,L),x(L))+
                SUM(R$DOWN(B,R),t(R));

MOMENT(B)..     SUM(R$UPP(B,R),DR(R)*t(R))=E=SUM(L$LOAD(B,L),
                DL(L)*x(L))+SUM(R$DOWN(B,R),DR(R)*t(R));

MODEL vigasycuerdas /COST,FORCES,MOMENT/;
SOLVE vigasycuerdas USING lp MAXIMIZING z;
```

La solución en formato GAMS es:

```
                       LOWER     LEVEL     UPPER    MARGINAL
---- VAR Z              -INF    640.000     +INF       .
  Z           valor de la funcion objetivo

---- VAR X           carga aplicada
      LOWER     LEVEL     UPPER    MARGINAL
L1     -INF    440.000     +INF       .
L2     -INF    200.000     +INF       .

---- VAR T           tension en la cuerda R
      LOWER     LEVEL     UPPER    MARGINAL
RA     -INF    240.000   300.000      .
RB     -INF    300.000   300.000     2.000
RC     -INF     25.000   200.000      .
RD     -INF     75.000   200.000      .
RE     -INF    100.000   100.000     0.400
RF     -INF    100.000   100.000      .
```

Por tanto, la solución es:

$$Z = 640; \quad x_1 = 440, \quad x_2 = 200$$

$$T_A = 240, \quad T_B = 300, \quad T_C = 25, \quad T_D = 75, \quad T_E = 100, \quad T_F = 100$$

11.2.7 El despacho económico de centrales térmicas

Este problema se estudió en el apartado 1.8, y en el ejemplo 1.7. Un ejemplo de este problema se formula como sigue. Minimizar

$$Z = \sum_{i=1}^{n} C_i \, p_i \tag{11.10}$$

sujeto a

$$\begin{array}{rcl} \delta_k & = & 0 \\ \sum_{j \in \Omega_i} B_{ij}(\delta_j - \delta_i) + p_i & = & D_i; \quad i = 1, 2, \ldots, n \\ -\overline{P}_{ij} \le B_{ij}(\delta_j - \delta_i) & \le & \overline{P}_{ij}; \quad \forall j \in \Omega_i, \; i = 1, 2, \ldots, n \\ \underline{P}_i \le p_i & \le & \overline{P}_i; \quad i = 1, 2, \ldots, n \end{array} \tag{11.11}$$

donde $n = 3$, $k = 3$

$$\underline{\mathbf{p}} = \begin{pmatrix} 0.15 \\ 0.10 \end{pmatrix}, \; \overline{\mathbf{p}} = \begin{pmatrix} 0.6 \\ 0.4 \end{pmatrix}, \; \mathbf{c} = \begin{pmatrix} 6 \\ 7 \end{pmatrix}, \; \mathbf{B} = \begin{pmatrix} 0.0 & 2.5 & 3.5 \\ 2.5 & 0.0 & 3.0 \\ 3.5 & 3.0 & 0.0 \end{pmatrix}, \; \mathbf{D} = \begin{pmatrix} 0.0 \\ 0.0 \\ 0.85 \end{pmatrix}$$

$$\mathbf{P} = \begin{pmatrix} 0.0 & 0.3 & 0.5 \\ 0.3 & 0.0 & 0.4 \\ 0.5 & 0.4 & 0.0 \end{pmatrix}, \ \Omega_1 = \{2,3\}, \ \ \Omega_2 = \{1,3\}, \ \ \Omega_3 = \{1,2\}$$

y las variables de optimización son p_1, p_2, δ_1, y δ_2.

A la hora de escribir el código GAMS correspondiente, se deben tener en cuenta las siguientes aclaraciones:

- Definición de un subconjunto en base a dos conjuntos distintos. Esto permite la asociación entre elementos de distintos conjuntos (generadores y nudos).
- El símbolo "*" se usa en algunas tablas para representar, por columnas, los datos de entrada de los generadores o de las líneas. El identificador que se asigna a dichas columnas es el que luego se usa en las restricciones para referirse a sus datos.
- Se emplea el operador condicional para sumar sólo los elementos que pertenecen al conjunto de interés.

Un fichero de entrada GAMS para este ejemplo se presenta a continuación.

```
$title EL PROBLEMA DEL DESPACHO ECONOMICO

**  Se definen los conjuntos G y N.
**  A continuacion se define el conjunto MAP como subconjunto de G y N.
**  El subconjunto MAP establece las combinaciones validas
**  entre los elementos de G y N.

SETS
  G   indice de generadores  /G1*G2/
  N   indice de nudos   /N1*N3/
  MAP(G,N) asocia generadores y nudos /G1.N1,G2.N2/;

**  Se asignan los datos de entrada. La primera tabla representa los datos
**  para cada generador. Los identificadores de las columnas no se han
**  declarado como elementos que pertenecen a algun conjunto,
**  por eso se utiliza el simbolo '*' en la definicion de la tabla.
**  Igualmente ocurre en la segunda tabla.

TABLE GDATA(G,*)   datos de entrada de los generadores
          PMIN     PMAX     COST
*         (kW)     (kW)    (E/kWh)
  G1      0.15     0.6       6
  G2      0.10     0.4       7;

TABLE LDATA(N,N,*) datos de entrada de las lineas
           SUS    LIMIT
*          (S)    (kW)
  N1.N2    2.5     0.3
  N1.N3    3.5     0.5
  N2.N3    3.0     0.4;

**  El siguiente parametro indica que el nudo 3 tiene asignada una demanda.
```

```
PARAMETER
  LOAD(N)  demanda en el nudo N
          / N3  0.85 /

**  Se declaran las variables de optimizacion.

VARIABLES
  z             valor de la funcion objetivo
  p(G)          potencia generada por el generador G
  d(N)          angulo en el nudo N;

**  Se limitan las variables con los datos definidos anteriormente.

  p.lo(G)=GDATA(G,'PMIN');
  p.up(G)=GDATA(G,'PMAX');

**  El nudo 3 se toma como nudo de referencia, por tanto,
**  se fija su angulo al valor cero.

  d.fx('N3')=0;

**  Se declaran las distintas restricciones.

EQUATIONS
  COST          funcion objetivo
  MAXPOW(N,N)   limita la capacidad maxima de la linea
  MINPOW(N,N)   limita la capacidad minima de la linea
  LOADBAL(N)    ecuacion de balance de potencias;

**  El conjunto de nudos N se duplica para hacer referencia
**  a distintos elementos del mismo dentro de la misma restriccion.


ALIAS(N,NP);

**  A continuacion se definen las restricciones.
**  La restriccion de la funcion objetivo se define como una suma.
**  El elemento 'COST' representa los costes de los generadores
**  dados por la columna 'COST' de la tabla GDATA.
**  A continuacion se definen las restricciones MAXPOW y MINPOW.
**  Como existen tantas restricciones como lineas, se definen
**  en funcion de los conjuntos de nudos N que estan conectados.
**  N y NP son identificadores del mismo conjunto de nudos N,
**  pero se refieren a elementos distintos del conjunto dentro de la
**  misma restriccion.
**  Finalmente, las restricciones LOADBAL se definen para cada nudo.
**  Para sumar solo la potencia de aquellos generadores G situados
**  en un nudo N, se utiliza la expresion condicional $MAP(G,N).

COST..         z =e= SUM(G,GDATA(G,'COST')*p(G));
MAXPOW(N,NP).. LDATA(N,NP,'SUS')*(d(N)-d(NP))=l= LDATA(N,NP,'LIMIT');
MINPOW(N,NP).. LDATA(N,NP,'SUS')*(d(N)-d(NP))=g=-LDATA(N,NP,'LIMIT');
LOADBAL(N)..   SUM(G$MAP(G,N),p(G))+SUM(NP,LDATA(N,NP,'SUS')*(d(N)-d(NP))+
               LDATA(NP,N,'SUS')*(d(N)-d(NP)))=e=LOAD(N);

**  Los comandos siguientes declaran y definen el modelo del despacho
**  economico, incluyendo todas las restricciones declaradas, y solicitan
```

```
**  a GAMS que resuelva el problema mediante un optimizador lineal.

MODEL de /COST,MAXPOW,MINPOW,LOADBAL/;
SOLVE de USING lp MINIMIZING z;
```

La solución de este problema (parte del fichero de salida GAMS) es:

```
---- VAR Z          valor de la funcion objetivo
      LOWER     LEVEL     UPPER    MARGINAL
      -INF      5.385     +INF        .

---- VAR P            potencia generada por el generador G
      LOWER     LEVEL     UPPER    MARGINAL
G1    0.150     0.565     0.600        .
G2    0.100     0.285     0.400        .

---- VAR D            angulo en el nudo N
      LOWER     LEVEL     UPPER    MARGINAL
N1    -INF     -0.143     +INF        .
N2    -INF     -0.117     +INF        .
N3      .         .         .        EPS
```

La solución óptima del problema del despacho económico es:

$$Z = 5.385, \qquad \mathbf{p} = (0.565, 0.285)^T, \qquad \boldsymbol{\delta} = (-0.143, -0.117, 0)^T$$

11.3 Ejemplos de programación lineal entera mixta

Este apartado está dedicado a la implementación en GAMS de los problemas de programación lineal entera mixta de capítulos anteriores.

11.3.1 El problema de la mochila

El problema de la mochila se estudió en detalle en el apartado 2.2. La formulación de este problema es como sigue.

Maximizar

$$Z = \sum_{j=1}^{n} c_j x_j$$

sujeto a

$$\begin{array}{rcl} \sum_{j=1}^{n} a_j x_j & \leq & b, \\ x_j & \in & \{0,1\} \quad \forall j = 1 \cdots n \end{array}$$

donde se supone que $a_j = c_j$, $b = 700$, y que

$$\mathbf{a} = (100, 155, 50, 112, 70, 80, 60, 118, 110, 55)^T$$

Como novedad, se tiene que en este tipo de problemas es conveniente especificar la precisión que se desea para la solución obtenida. En este ejemplo, se especifica una precisión de 10^{-10} (`OPTION OPTCR=1e-10`).

Un fichero de entrada GAMS que resuelve este problema es:

```
$title EL PROBLEMA DE LA MOCHILA.

** El problema de la mochila determina cual es la carga maxima
** que puede llevar un excursionista en su mochila.
** En el ejemplo implementado ocurre que A(J)=C(J).
** Esta igualdad se tiene en cuenta para reducir el codigo
** GAMS, pero el programa es generico para cualquier
** problema tipo mochila.

** El comando siguiente modifica el criterio de parada
** del optimizador lineal entero mixto. Si no se especifica
** esta precision, el optimizador se detiene encontrando
** una solucion de valor de la funcion objetivo menor que la deseada.

OPTION OPTCR=1e-10;

SET
 J    conjunto de contenedores   /c1*c10/;

** Los vectores de datos A y C se definen como parametros.

PARAMETERS
 C(J) beneficio del contenedor J
     /c1     100
      c2     155
      c3      50
      c4     112
      c5      70
      c6      80
      c7      60
      c8     118
      c9     110
      c10     55/
 A(J) peso del contenedor J;
 A(J) = C(J);

SCALAR  B    capacidad maxima del carguero /700/;

**  Se declaran las variables de optimizacion.

VARIABLES
 z      valor de la funcion objetivo
 x(J)   decision de carga;

** x(J) vale 1 si se carga el contenedor y 0 si no se carga.

BINARY VARIABLE x;

** Se declaran las restricciones.

EQUATIONS
 COST funcion objetivo
```

```
 CAPA limita la capacidad del carguero;

** La funcion objetivo es la suma de los pesos de los contenedores que se cargan.
** Las restricciones CAPA aseguran que el total del peso de la carga no exceda
** la capacidad maxima del carguero.

COST ..  z=E= SUM(J,C(J)*x(J));
CAPA ..       SUM(J,A(J)* x(J)) =L= B;

**  Los comandos siguientes declaran y definen el modelo de la mochila,
**  incluyendo todas las restricciones declaradas, y solicitan
**  a GAMS que resuelva el problema mediante un optimizador entero mixto.

MODEL mochila /ALL/;
SOLVE mochila USING mip MAXIMIZING z;
```

Parte del fichero de salida GAMS es:

```
                       LOWER     LEVEL     UPPER    MARGINAL
---- VAR Z              -INF    700.000     +INF       EPS
  Z          valor de la funcion objetivo

---- VAR X           decision de carga
       LOWER     LEVEL     UPPER    MARGINAL
c1       .        1.000     1.000      EPS
c2       .          .       1.000      EPS
c3       .        1.000     1.000      EPS
c4       .        1.000     1.000      EPS
c5       .        1.000     1.000      EPS
c6       .        1.000     1.000      EPS
c7       .        1.000     1.000      EPS
c8       .        1.000     1.000      EPS
c9       .        1.000     1.000      EPS
c10      .          .       1.000      EPS
```

La solución obtenida indica que sólo se cargan los contenedores

$$c_1, c_3, c_4, c_5, c_6, c_7, c_8, c_9.$$

El valor óptimo de la función objetivo es $Z = 700$ toneladas, por lo que el carguero va lleno.

11.3.2 La identificación de síntomas relevantes

Este problema se estudió en el apartado 2.3. La formulación de este problema de optimización es como sigue.

Minimizar

$$Z = \sum_{j=1}^{m} x_j$$

sujeto a

$$\sum_{j=1}^{m} x_j d(c_{ij} - c_{kj}) > a; \quad k \in \{1, 2, \dots, n\}, \ i \neq k \tag{11.12}$$

donde

$$d(c_{ij} - c_{kj}) = \begin{cases} 1 & if \quad c_{ij} \neq c_{kj} \\ 0 & if \quad c_{ij} = c_{kj} \end{cases} \tag{11.13}$$

El fichero de entrada GAMS que resuelve este problema es:

```
$Title SINTOMAS SIGNIFICATIVOS

SETS   D conjunto de enfermedades /D1*D5/
       S conjunto de sintomas /S1*S8/;
ALIAS(D,DP);

SCALAR A nivel de discrepancia /1/;
TABLE  C(D,S) nivel del sintoma S para la enfermedad D
       S1   S2   S3   S4   S5   S6   S7   S8
 D1     2    3    1    1    1    2    1    2
 D2     1    1    1    1    3    1    2    1
 D3     3    4    2    3    2    2    3    2
 D4     2    2    2    2    2    1    2    3
 D5     1    1    1    2    1    1    1    2;
PARAMETER
DD(D,DP,S) medida de la discrepancia;
DD(D,DP,S)$(C(D,S) ne C(DP,S))=1;

BINARY VARIABLE x(S);
FREE VARIABLE z;
EQUATIONS
 SELECT     numero de sintomas seleccionados
 Suff(D,DP) los sintomas identifican a la enfermedad;

SELECT.. z =e= SUM(S,x(S));
Suff(D,DP)$(ord(D) ne ord(DP))..
             SUM(S,x(S)*DD(D,DP,S)) =g= A;

MODEL sintomas /ALL/;

SOLVE sintomas USING mip MINIMIZING z;
```

La solución obtenida indica que bastan los síntomas 1 y 4 para identificar todas las enfermedades.

11.3.3 El problema de la academia de ingeniería

Este problema se estudió en el apartado 2.3. El problema de optimización se formula como sigue.

Maximizar y minimizar

$$Z_j = \sum_{i=1}^{I} \sum_{s=1}^{S} x_{ijs}, \quad \forall j \in \{1, 2, \ldots, J\}$$

sujeto a

$$
\begin{array}{rcll}
\displaystyle\sum_{s=1}^{S} x_{ijs} & \leq & 1, & \forall i \in \{1,2,\ldots,I\},\ j \in \{1,2,\ldots,J\} \\
\displaystyle\sum_{j=1}^{J} x_{ijs} & \leq & 1, & \forall i \in \{1,2,\ldots,I\},\ s \in \{1,2,\ldots,S\} \\
\displaystyle\sum_{i=1}^{I}\sum_{s=1}^{S} p_s x_{ijs} & = & c_j, & \forall j \in \{1,2,\ldots,J\} \\
x_{ijs} & \in & \{0,1\}, & \forall i \in \{1,2,\ldots,I\},\ j \in \{1,2,\ldots,J\},\ s \in \{1,2,\ldots,S\}
\end{array}
$$

Para resolver este problema mediante GAMS se utilizan dos estructuras nuevas de GAMS: bucles y conjuntos dinámicos. Es necesario resolver dos problemas para cada candidato, que solo difieren en la dirección de la optimización. Se utiliza un bucle controlado por el índice de los candidatos. Dentro del bucle, se actualiza un conjunto dinámico según el candidato de que se trate. La restricción de la función objetivo se define mediante el conjunto dinámico, de forma que en cada iteración este conjunto es distinto. En el código GAMS se han insertado comentarios adicionales para clarificar estas novedades.

Un fichero de entrada GAMS que resuelve este problema es:

```
$title EL PROBLEMA DE LA ACADEMIA

SETS
 I numero de miembros /1*20/
 J numero de candidatos /1*8/
 DIN(J) conjunto dinamico
 S numero de puntuaciones distintas que pueden asignarse /1*4/;

ALIAS(J,J1);

PARAMETER
P(S) puntuaciones S
 /1 10
  2 8
  3 3
  4 1/;

TABLE N(I,J) puntuacion que recibe el candidato J del miembro I
        1       2       3       4       5       6       7       8
********************************************************************
1       3               10              8       1
2       1               10              8       3
3               1               3       10              8
4               3       10              8       1
5       3               8               10              1
6       1               10              8               3
7       10              8               3       1
8       3               10      1       8
9       8               3               10      1
10              3       10              1               8
```

```
11      8               1               10      3
12                                      10
13                      10              8
14      10                      1       3                       8
15      3               10              8               1
16      10              1               8               3
17      1       3       10      8
18      1       3       8               10
19      1               10              3       8
20      8       1       10              3 ;

SCALARS zmin,zmax;

PARAMETER C(J) puntuacion total del candidato J;
 C(J)=sum(I,N(I,J));

VARIABLES
 z valor de la funcion objetivo;

BINARY VARIABLE
 x(I,J,S) vale 1 si el miembro I asigna la puntuacion P(S) al candidato J,
          si no vale 0.;

EQUATIONS
 OBJ funcion a optimizar
 L1(I,J) Cada miembro puede asignar como mucho una puntuacion a cada candidato
 L2(I,S) Cada miembro puede asignar una puntuacion S como mucho a un candidato
 TOTALSCORE(J) puntuacion total de cada candidato J;

** La restriccion de la funcion objetivo varia segun el candidato.
** Para cada candidato, la funcion a optimizar (minimizar o maximizar)
** es el numero de miembros que asignan una puntuacion a dicho candidato.
** El conjunto dinamico DIN permite resolver el mismo problema
** para distintos candidatos.
** Este conjunto se actualiza en cada iteracion dentro del bucle.
** Las restricciones son las mismas para cada candidato.

OBJ(J)$DIN(J)..z=e=sum(I,sum(S,x(I,J,S)));
L1(I,J)..sum(S,x(I,J,S))=l=1;
L2(I,S)..sum(J,x(I,J,S))=l=1;
TOTALSCORE(J)..sum(I,sum(S,P(S)*x(I,J,S)))=e=C(J);

**  El modelo incluye todas las restricciones.
MODEL Academia /ALL/;

** Se abre el fichero de salida.
file aux /academia.out/;
put aux;

** Primero, se inicia el conjunto dinamico al conjunto vacio
DIN(J)=NO;

loop(J1,

** Solo se considera un candidato en cada iteracion.
      DIN(J1)=YES;
```

```
** Se minimiza el problema para el candidato J1.
      Solve Academia using mip Minimizing z;

** La solucion se guarda en zmin.
      zmin=z.l;

** Se maximiza el problema para el candidato J1.
      Solve Academia using mip Maximizing z;

** La solucion se guarda en zmax.
      zmax=z.l;

** Las dos soluciones se escriben en el fichero 'academia.out'.
      put "J=",J1.tl:3," zmin= ",zmin:3:0," zmax= ",zmax:3:0/;

** Una vez optimizado el problema en las dos direcciones, el conjunto
** dinamico del candidato considerado se actualiza con el conjunto vacio.
      DIN(J1)=NO;
);
```

y el contenido del fichero de salida es

```
J=1   zmin=  8 zmax= 20
J=2   zmin=  3 zmax= 14
J=3   zmin= 15 zmax= 20
J=4   zmin=  2 zmax= 13
J=5   zmin= 15 zmax= 20
J=6   zmin=  2 zmax= 18
J=7   zmin=  3 zmax= 20
J=8   zmin=  1 zmax=  8
```

De la solución obtenida se pueden extraer las siguientes conclusiones:

1. No es necesario especificar la tabla $N(I, J)$, pues se supone que las puntuaciones asignadas a los candidatos no se conocen. Sin embargo, en este ejemplo se han utilizado para obtener los valores de $C(J)$.

2. Los límites (superior e inferior) de puntuación para los candidatos con mayor puntuación son menores que los límites para los candidatos de menor puntuación.

3. Aunque el candidato 6 tiene una puntuación máxima de 18 y ésta es mayor que la correspondiente al candidato 2 (puntuación de 14), la puntuación mínima del candidato 6 (2) es menor que la del candidato 2 (3).

4. No se asegura la selección de los candidatos 3 y 5, pues las puntuaciones máximas de los candidatos 1, 6, y 7 (20, 18 y 20, respectivamente) son mayores que la puntuación mínima (15) de estos dos candidatos.

11.3.4 El problema del horario

Este problema se estudió en el apartado 2.5. Se formula como sigue.

Minimizar

$$\sum_{s\in\Omega}\sum_{c=1}^{n_c}\sum_{h=1}^{n_h}(c+h)\ v(s,c,h)$$

sujeto a

$$\begin{array}{rcl}\sum_{s\in\Omega_i}\sum_{c=1}^{n_c}\sum_{h=1}^{n_h}v(s,c,h) & = & n_i,\ \ \forall i\\ \sum_{s\in\Omega_i}\sum_{c=1}^{n_c}v(s,c,h) & \leq & 1,\ \ \forall h,\ \ \forall i\\ \sum_{c=1}^{n_c}\sum_{h=1}^{n_h}v(s,c,h) & = & 1,\ \ \forall s\\ \sum_{s\in\Omega}v(s,c,h) & \leq & 1,\ \ \forall c,\ \ \forall h\\ \sum_{s\in\Delta_b}\sum_{c=1}^{n_c}v(s,c,h) & \leq & 1,\ \ \forall h,\ \ \forall b\end{array}$$

Un fichero de entrada GAMS para este problema es:

```
$title EL PROBLEMA DEL HORARIO

SETS
 C   aulas        /c1*c3/
 H   horas        /h1*h5/
 S   asignaturas  /s1*s8/
 I   profesores   /i1,i2/
 B   cursos       /b1,b2/
 SI(S,I) asociacion de asignaturas y profesores /(s1,s2,s8).i1,(s3*s7).i2/
 SB(S,B) asociacion de asignaturas y cursos     /(s1*s4).b1,(s5*s8).b2/;

VARIABLE z;
BINARY VARIABLE v(S,C,H);

EQUATIONS
 coste        compacta el horario
 const1(I)    cada profesor imparte todas sus asignaturas
 const2(H,I)  cada profesor imparte como mucho 1 asignatura en cada hora
 const3(S)    cada asignatura se imparte una sola vez
 const4(C,H)  en cada clase y hora se imparte como mucho 1 asignatura
 const5(H,B)  en cada hora se imparte como mucho 1 asignatura de cada curso;

 coste..        SUM((S,C,H),(ord(C)+ord(H))*v(S,C,H))=e=z ;
 const1(I)..    SUM((S,C,H)$SI(S,I),v(S,C,H))=e=SUM(S$SI(S,I),1);
 const2(H,I)..  SUM((S,C)$SI(S,I),v(S,C,H))=l=1;
 const3(S)..    SUM((C,H),v(S,C,H))=e=1;
 const4(C,H)..  SUM(S,v(S,C,H))=l=1;
 const5(H,B)..  SUM((S,C)$SB(S,B),v(S,C,H))=l=1;

model horario /all/;
solve horario using mip minimizing z;

DISPLAY v.L;
```

Parte del fichero de salida GAMS es:

```
                      LOWER     LEVEL     UPPER    MARGINAL
```

```
---- VAR Z              -INF     32.000     +INF       .

----     34 VARIABLE  V.L
              H1          H2          H3          H4          H5
S1.C2                    1.000
S2.C2       1.000
S3.C1                                1.000
S4.C1                                            1.000
S5.C1                                                        1.000
S6.C1                    1.000
S7.C1       1.000
S8.C2                                1.000
```

La solución con formato de tabla se presentó en el apartado 2.5.

11.3.5 Modelos de localización de plantas productivas

Este problema se estudió en el apartado 2.6. Su formulación es la siguiente.

Maximizar

$$Z = \sum_{i \in I} \sum_{j \in J} c_{ij} x_{ij} - \sum_{j \in J} f_j y_j$$

sujeto a

$$\begin{array}{rcll} \sum_{j \in J} x_{ij} & = & b_i, & \forall i \in I \\ \sum_{i \in I} x_{ij} & \leq & u_j y_j, & \forall j \in J \\ y_j & \in & \{0,1\}, & \forall j \in J \\ x_{ij} & \geq & 0, & \forall i \in I, \forall j \in J \end{array}$$

donde se supone que $u_j = 6\ \forall j$, $f_j = 10\ \forall j$ y

$$\mathbf{b} = (1.5, 2.0, 3.0, 4.0, 2.5, 1.0, 2.0)$$

Un fichero de entrada en GAMS para este ejemplo es:

```
$Title EL PROBLEMA DE LA LOCALIZACION DE PLANTAS PRODUCTIVAS

** El comando siguiente modifica la precision con la que se obtiene una
** solucion entera mediante un criterio relativo. Si no se indica, GAMS
** finaliza con un valor de la funcion objetivo menor que el maximo posible.

OPTION OPTCR=1e-10;

** En primer lugar se definen los conjuntos.
** El conjunto I se refiere a las 7 ciudades.
** El conjunto J se refiere a las 6 posibles
** localizaciones para las naves industriales.

SET
 I indice de ciudades       /C1*C7/
 J indice de localizaciones /L1*L6/;

PARAMETERS
```

```
 B(I) demanda del producto en la ciudad I
  /C1   1.5
   C2   2.0
   C3   3.0
   C4   4.0
   C5   2.5
   C6   1.0
   C7   2.0/

 F(J) coste de amortizacion de una nave industrial situada en J
 U(J) capacidad maxima de produccion de una nave industrial en J;

 F(J) = 10;
 U(J) = 6;

** A continuacion se muestran los datos de entrada. La tabla
** contiene los beneficios de producir en una nave situada
** en J, y vender el producto en la ciudad I.

TABLE  C(J,I) beneficios segun las distintas localizaciones
        C1      C2      C3      C4      C5      C6     C7
 L1      4.0     4.5     2.5     0.5     1.0     0.5    -3.5
 L2      4.0     4.5     2.5     4.2     3.5     1.5    -0.5
 L3      3.5     5.0     4.0     3.5     4.5     1.5     0.0
 L4      1.3     3.0     5.0     3.3     5.5     1.8     1.3
 L5      0.5     1.0     1.5     5.0     4.0     5.5     3.0
 L6     -1.0     0.0     1.5     3.3     4.0     4.5     2.0 ;

** Se declaran las variables de optimizacion.

VARIABLES
 z      valor de la funcion objetivo
 x(I,J) cantidad producida en J y vendida a I
 y(J)   variable de localizacion;

** y(J) vale 1 si la nave se construye en J, y 0 en otro caso.

** Las variables y(J) son declaradas como binarias y las
** variables x(I,J) como positivas.

POSITIVE VARIABLE x;
BINARY VARIABLE y;

** Se declaran las restricciones.

EQUATIONS
 COST    funcion objetivo
 SD(I)   se satisface la demanda de la ciudad I
 CAPA(J) capacidad de produccion de la nave J;

** Se definen las restricciones del problema.
** La funcion objetivo es la suma de los beneficios
** menos los costes de inversion.
** La restriccion SD(I) asegura que se cumpla la demanda de I.
** La restriccion CAPA(J) asegura que la produccion en la
** nave no supere su capacidad maxima.
```

```
COST   .. z=e= SUM((I,J),C(J,I)*x(I,J))-SUM(J,F(J)*y(J));
SD(I)  .. SUM(J,x(I,J)) =e= B(I);
CAPA(J).. SUM(I,x(I,J)) =l= U(J)*y(J);

**  Los comandos siguientes declaran y definen el modelo,
**  incluyendo todas las restricciones declaradas, y solicitan
**  a GAMS que resuelva el problema mediante un optimizador entero mixto.

MODEL loc /all/;
SOLVE loc USING mip MAXIMIZING z;
DISPLAY x.l;
```

La solución de este problema (parte del fichero de salida GAMS) viene dada por:

```
                      LOWER     LEVEL     UPPER    MARGINAL
---- VAR Z             -INF     44.450     +INF       EPS
  Z          valor de la funcion objetivo

---- VAR Y          variable de localizacion
      LOWER     LEVEL     UPPER    MARGINAL
l1      .         .       1.000   -10.000
l2      .       1.000     1.000   -10.000
l3      .         .       1.000    -7.000
l4      .       1.000     1.000   -10.000
l5      .       1.000     1.000    -5.200
l6      .         .       1.000   -10.000

----     56 VARIABLE  X.L
            l2          l4          l5
c1       1.500
c2       2.000
c3                   3.000
c4       1.000                   3.000
c5                   2.500
c6                               1.000
c7                               2.000
```

La interpretación del valor óptimo de este problema es que se deben instalar 3 naves industriales en los lugares L_2, L_4, y L_5, y que el suministro a las ciudades se distribuye como sigue:

	Ciudades						
Localizaciones	C_1	C_2	C_3	C_4	C_5	C_6	C_7
L_2	1.5	2.0	–	1.0	–	–	–
L_4	–	–	3.0	–	2.5	–	–
L_5	–	–	–	3.0	–	1.0	2.0

11.3.6 Programación horaria de centrales térmicas

El problema de la programación horaria se estudió en el apartado 2.7, y en el ejemplo 2.6. La formulación de este problema es como sigue.

Minimizar

$$Z = \sum_{k=1}^{K} \sum_{j=1}^{J} [A_j \; v_{jk} + B_j \; p_{jk} + C_j \; y_{jk} + D_j \; z_{jk}] \qquad (11.14)$$

sujeto a

$$\begin{array}{rcll} \underline{P}_j v_{jk} \le p_{jk} & \le & \overline{P}_j v_{jk}; & \forall j, \forall k \\ p_{jk+1} - p_{jk} & \le & S_j; & \forall j, k = 0, \cdots, K-1 \\ p_{jk} - p_{jk+1} & \le & T_j; & \forall j, k = 0, \cdots, K-1 \\ y_{jk} - z_{jk} & = & v_{jk} - v_{jk-1}; & \forall j, k = 1, \cdots, K \\ \sum_{j=1}^{J} p_{jk} & = & D(k); & \forall k \\ \sum_{j=1}^{J} \overline{P}_j \; v_{jk} & \ge & D(k) + R(k); & \forall k \end{array} \qquad (11.15)$$

donde $K = 4$, $J = 3$, y

$$\underline{\mathbf{P}} = \begin{pmatrix} 50 \\ 80 \\ 40 \end{pmatrix}, \quad \overline{\mathbf{P}} = \begin{pmatrix} 350 \\ 200 \\ 140 \end{pmatrix}, \quad \mathbf{T} = \begin{pmatrix} 300 \\ 150 \\ 100 \end{pmatrix}, \quad \mathbf{S} = \begin{pmatrix} 200 \\ 100 \\ 100 \end{pmatrix}$$

$$\mathbf{A} = \begin{pmatrix} 5 \\ 7 \\ 6 \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} 20 \\ 18 \\ 5 \end{pmatrix}, \quad \mathbf{C} = \begin{pmatrix} 0.5 \\ 0.3 \\ 1.0 \end{pmatrix}, \quad \mathbf{E} = \begin{pmatrix} 0.100 \\ 0.125 \\ 0.150 \end{pmatrix}$$

$$\mathbf{D} = \begin{pmatrix} 150 \\ 500 \\ 400 \end{pmatrix}, \quad \mathbf{R} = \begin{pmatrix} 15 \\ 50 \\ 40 \end{pmatrix}$$

Téngase en cuenta la siguiente aclaración para entender el código GAMS para este ejemplo:

- Se necesitan expresiones condicionales en las restricciones, de forma que la condición `$(ord(K) GT 1)` se incluye en las restricciones para indicar que éstas se deben cumplir para todos los períodos excepto para el período inicial.

Un fichero de entrada GAMS que resuelve este problema es:

```
$title EL PROBLEMA DE LA PROGRAMACION HORARIA

** Primero se declaran los conjuntos.

SETS
  K indice de periodos de tiempo /1*4/
  J indice de generadores /1*3/


**  Se asignan los datos de entrada. La primera tabla representa los datos
**  para cada generador. Los identificadores de las columnas no se han
**  declarado como elementos que pertenecen a algun conjunto,
**  por eso se utiliza el simbolo '*' en la definicion de la tabla.
```

```
**  Igualmente ocurre en la segunda tabla.


TABLE GDATA(J,*) datos de entrada de los generadores
        PMIN   PMAX    T      S     A     B     C      E
*       (kW)   (kW)  (kW/h) (kW/h) (E)   (E)   (E)  (E/kWh)
  1      50    350    300   200     5    20    0.5   0.100
  2      80    200    150   100     7    18    0.3   0.125
  3      40    140    100   100     6     5    1.0   0.150;

TABLE PDATA(K,*) datos de demanda por periodo
       D     R
*     (kW) (kW)
  2   150   15
  3   500   50
  4   400   40;

**  Se declaran las variables de optimizacion.

VARIABLES
  z      valor de la funcion objetivo
  p(J,K) potencia generada por el generador j en el periodo k
  v(J,K) vale 1 si el generador j esta acoplado en el periodo k
  y(J,K) vale 1 si el generador j se arranca al comienzo de k
  s(J,K) vale 1 si el generador j se para al comienzo del periodo k;

**  La potencia es una variable positiva.

POSITIVE VARIABLES p(J,K);

**  Las decisiones de acoplamiento se tratan como variables binarias.

BINARY VARIABLES v(J,K),y(J,K),s(J,K);

**  Se asignan valores iniciales a algunas variables
  v.fx(J,'1')=0;
  p.fx(J,'1')=0;

** Se declaran las restricciones

EQUATIONS
  COST          funcion objetivo
  PMAXLIM(J,K)  maximo de la potencia generada
  PMINLIM(J,K)  minimo de la potencia generada
  LOAD(K)       balance de potencias
  RESERVE(K)    reserva rodante
  LOGIC(J,K)    logica de arranques y paradas
  RAMPUP(J,K)   rampa de subida
  RAMPDOWN(J,K) rampa de bajada;

COST.. z =e= SUM((K,J), GDATA(J,'A')*v(J,K)+GDATA(J,'B')*y(J,K)+
                        GDATA(J,'C')*s(J,K)+GDATA(J,'D')*p(J,K));
PMAXLIM(J,K)$(ord(K) GT 1).. p(J,K)=l=GDATA(J,'PMAX')*v(J,K);
PMINLIM(J,K)$(ord(K) GT 1).. p(J,K)=g=GDATA(J,'PMIN')*v(J,K);
LOAD(K)$(ord(K) GT 1)..      SUM(J,p(J,K))=e=PDATA(K,'D');
RESERVE(K)$(ord(K) GT 1)..   SUM(J,GDATA(J,'PMAX')*v(J,K))=g=PDATA(K,'D')
                             +PDATA(K,'R');
```

```
LOGIC(J,K)$(ord(K) GT 1)..   y(J,K)-s(J,K)=e=v(J,K)-v(J,K-1);
RAMPUP(J,K)$(ord(K) GT 1)..  p(J,K)-p(J,K-1)=l=GDATA(J,'S');
RAMPDOWN(J,K)$(ord(K) GT 1)..p(J,K-1)-p(J,K)=l=GDATA(J,'T');

**  Los comandos siguientes declaran y definen el modelo de programacion
**  horaria, incluyendo todas las restricciones declaradas, y solicitan
**  a GAMS que resuelva el problema mediante un optimizador entero mixto.

MODEL ph /ALL/;
SOLVE ph USING mip MINIMIZING z;
```

Parte del fichero de salida GAMS, con la solución óptima, es la siguiente.

```
    LOWER     LEVEL     UPPER    MARGINAL
---- VAR Z
    -INF    191.000     +INF       .
    Z          valor de la funcion objetivo

---- VAR P           potencia generada por el generador j en el periodo k
       LOWER     LEVEL     UPPER    MARGINAL
1.0      .         .         .        EPS
1.1      .      150.000     +INF       .
1.2      .      350.000     +INF       .
1.3      .      320.000     +INF       .
2.0      .         .         .        EPS
2.1      .         .        +INF      EPS
2.2      .      100.000     +INF       .
2.3      .       80.000     +INF       .
3.0      .         .         .        EPS
3.1      .         .        +INF     0.050
3.2      .       50.000     +INF       .
3.3      .         .        +INF     0.050

---- VAR V  vale 1 si el generador j se acopla en el periodo k
       LOWER     LEVEL     UPPER    MARGINAL
1.0      .         .         .        EPS
1.1      .        1.000     1.000     5.000
1.2      .        1.000     1.000   -12.500
1.3      .        1.000     1.000     5.000
2.0      .         .         .        EPS
2.1      .         .        1.000     7.000
2.2      .        1.000     1.000     7.000
2.3      .        1.000     1.000     9.000
3.0      .         .         .        EPS
3.1      .         .        1.000     6.000
3.2      .        1.000     1.000     6.000
3.3      .         .        1.000     6.000

---- VAR Y vale 1 si el generador j se arranca al comienzo de k
                   beginning de periodo k
       LOWER     LEVEL     UPPER    MARGINAL
1.1      .       1.000     1.000    20.000
1.2      .         .       1.000    20.000
1.3      .         .       1.000    20.000
2.1      .         .       1.000    18.000
2.2      .       1.000     1.000    18.000
2.3      .         .       1.000    18.000
3.1      .         .       1.000     5.000
```

```
3.2      .       1.000     1.000     5.000
3.3      .         .       1.000     5.000

---- VAR S vale 1 si el generador j se para en el periodo k
       LOWER     LEVEL     UPPER    MARGINAL
1.1      .         .        1.000     0.500
1.2      .         .        1.000     0.500
1.3      .         .        1.000     0.500
2.1      .         .        1.000     0.300
2.2      .         .        1.000     0.300
2.3      .         .        1.000     0.300
3.1      .         .        1.000     1.000
3.2      .         .        1.000     1.000
3.3      .       1.000      1.000     1.000
```

Por tanto, la solución óptima es

$$Z = 191, \quad \mathbf{p} = \begin{pmatrix} 150 & 350 & 320 \\ 0 & 100 & 80 \\ 0 & 50 & 0 \end{pmatrix}, \quad \mathbf{v} = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 1 & 0 \end{pmatrix}$$

$$\mathbf{y} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \end{pmatrix}, \qquad \mathbf{s} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

11.4 Ejemplos de programación no lineal

En este apartado se implementan en GAMS los problemas de programación no lineal que se estudiaron en el capítulo 3.

11.4.1 El ejemplo del paquete postal

El problema del paquete postal se estudió en detalle en el apartado 3.2.1. La formulación de este problema es como sigue.

Maximizar

$$Z = xyz$$

sujeto a

$$z + 2x + 2y \leq 108$$

$$x, y, z \geq 0$$

Un fichero de entrada GAMS que resuelve este problema es

```
$title PROBLEMA DEL PAQUETE POSTAL

POSITIVE VARIABLES x,y,z;
FREE VARIABLE obj;

EQUATIONS
 VOL funcion objetivo
 DIM restriccion de dimension y peso;
```

```
VOL.. obj =e= x*y*z;
DIM.. z+2*x+2*y =l= 108;

MODEL paquete /ALL/;

x.l=1; y.l=1; z.l=1;

SOLVE paquete USING nlp MAXIMIZING obj;
```

La solución de este problema es:

$$Z = 11664; \quad x = 18; \quad y = 18; \quad z = 36$$

11.4.2 El ejemplo de la tienda

El problema de la tienda se estudió detalladamente en el apartado 3.2.2. La formulación de este problema es como sigue.

Minimizar

$$Z = 4(2ab + a\sqrt{h^2 + a^2})$$

sujeto a

$$\begin{array}{rcl} V & = & 4a^2(b + h/3) \\ H & = & b + h \\ a, b, c & \geq & 0 \end{array}$$

Un fichero de entrada GAMS para resolver este problema es:

```
$title EL PROBLEMA DE LA TIENDA

SCALAR TV  volumen total de la tienda /50/
       TH  altura total de la tienda  /3/;

POSITIVE VARIABLES a,b,h;
FREE VARIABLE z;

EQUATIONS
 SURFACE funcion objetivo
 VOLUME  restriccion de volumen de la tienda
 HEIGHT  restriccion de altura de la tienda;

SURFACE.. z =e= 4*(2*a*b+a*SQRT(SQR(h)+SQR(a)));
VOLUME..  TV =e= 4*SQR(a)*(b+(h/3));
HEIGHT..  TH =e= b+h;

MODEL tienda /ALL/;

a.l=100; b.l=100; h.l=100;
SOLVE tienda USING nlp MINIMIZING z;
```

La solución de este problema es:

$$Z = 58.937; \quad a = 2.633; \quad b = 1.205; \quad h = 1.795$$

11.4.3 El ejemplo de la lámpara

El problema de la lámpara se estudió en el apartado 3.2.3. La formulación para este problema es como sigue.

Minimizar

$$Z = k\frac{h^{1/2}}{(h^2+r^2)^{3/4}}$$

sujeto a

$$h \geq 0$$

Un fichero de entrada GAMS para resolver el problema es:

```
$title PROBLEMA DE LA LAMPARA

SCALAR k constante de proporcionalidad /3/
       r radio  /30/;

POSITIVE VARIABLE h;
FREE VARIABLE z;

EQUATION
 INTENSITY funcion objetivo;

INTENSITY.. z =e= k*(SQRT(h)/((SQR(h)+SQR(r))**0.75));

MODEL bombilla /all/;

h.l=10;
SOLVE bombilla USING nlp MAXIMIZING z;
```

La solución del problema es:

$$Z = 0.062;\ \ h = 21.183$$

11.4.4 El ejemplo de la superficie

El problema de la superficie se estudió en el apartado 3.2.4. La formulación de este problema es como sigue.

Minimizar

$$Z = \sqrt{x^2+y^2+z^2}$$

sujeto a

$$\begin{array}{rcl} xyz & = & 1 \\ x,y,z & \geq & 0 \end{array}$$

Un fichero de entrada GAMS para resolver este problema es

```
$title PROBLEMA DE LA SUPERFICIE

POSITIVE VARIABLES x,y,z;
FREE VARIABLE obj;

EQUATIONS
 DIST funcion objetivo
```

```
 ON   restricciones sobre los puntos;

DIST.. obj =e= SQRT(SQR(x)+SQR(y)+SQR(z));
ON..   1 =e= x*y*x;

MODEL superficie /all/;

x.l=100; y.l=100; z.l=100;
SOLVE superficie USING nlp MINIMIZING obj;
```

La solución del problema es:

$$Z = 1.732; \;\; x = 1; \;\; y = 1; \;\; z = 1$$

11.4.5 El ejemplo del transporte de arena

El problema del transporte de arena se estudió en el apartado 3.2.5. La formulación de este problema es como sigue.

Minimizar

$$Z = k(3(2xy) + 2(2xz) + 2yz) + 2\frac{50}{xyz}$$

sujeto a

$$x, y, z \;\; \geq \;\; 0$$

Un fichero de entrada GAMS que resuelve este problema es:

```
$title PROBLEMA DEL TRANSPORTE DE ARENA

SCALAR k constante de proporcionalidad /1.5/
       cu medidas cubicas de arena     /50/;

POSITIVE VARIABLES x,y,z;
FREE VARIABLE obj;

EQUATIONS
 COST funcion objetivo;

COST.. obj =e= k*(3*(2*x*y)+2*(2*x*z)+2*y*z)+2*(cu/(x*y*z));

MODEL arena /COST/;

x.l=0.5; y.l=0.5; z.l=0.5;

SOLVE arena USING nlp MINIMIZING obj;
```

La solución del problema es:

$$Z = 66.140; \;\; x = 0.857; \;\; y = 1.715; \;\; z = 2.572$$

11.4.6 El ejemplo del voladizo

El problema del voladizo se estudió en el apartado 3.3.1. La formulación de este problema es como sigue.

Minimizar

$$Z = \gamma Lxy$$

sujeto a

$$\begin{array}{lcl} \dfrac{4FL^3}{Exy^3} & \leq & S \\ x & \geq & 0.5 \\ x, y & \geq & 0 \end{array}$$

Un fichero de entrada GAMS para resolver este problema es

```
$title PROBLEMA DEL VOLADIZO

SCALARS L longitud /1/
        E modulo de Young /1e6/
        F carga en el extremo libre /100/
        S maxima deformacion permitida /1 /
        gamma densidad /100/;

POSITIVE VARIABLES x,y;
FREE VARIABLE obj;

EQUATIONS
 WEIGHT funcion objetivo
 SMT restriccion impuesta por la teoria de la resistencia de los materiales;

WEIGHT.. obj =e= gamma*L*x*y;
SMT..    (4*F*POWER(L,3))/(E*x*POWER(y,3))=l= S;

MODEL voladizo /ALL/;
x.lo=0.001; y.lo=0.001;
x.l=1; y.l=1;

SOLVE voladizo USING nlp MINIMIZING obj;
```

La solución de este problema es:

$$Z = 4.642; \quad x = 0.5; \quad y = 0.093$$

11.4.7 El ejemplo de la estructura con dos barras

El problema de la estructura con dos barras se estudió en el apartado 3.3.2. La formulación del problema es como sigue.

Minimizar

$$Z = 2\gamma\sqrt{x^2 + h^2}z$$

sujeto a

$$
\begin{array}{rcl}
D(x,z) & = & \dfrac{F}{Eh^2 2\sqrt{2}} \dfrac{(h^2+x^2)^{3/2}(h^4+x^4)^{1/2}}{x^2 z} \le D_0 \\
S^1(x,z) & = & \dfrac{F}{2\sqrt{2}h} \dfrac{(x+h)\sqrt{x^2+h^2}}{xz} \le S_0 \\
S^2(x,z) & = & \dfrac{F}{2\sqrt{2}h} \dfrac{(h-x)\sqrt{x^2+h^2}}{xz} \le S_0 \\
x,z & \ge & 0
\end{array}
$$

Un fichero de entrada GAMS que resuelve este problema es:

```
$title PROBLEMA DE LA ESTRUCTURA CON 2 BARRAS

SCALARS Gamma  densidad del material de las barras /1e2/
        E      modulo de Young /1e6/
        F      carga sobre el pivote fijo /15e3/
        S0     tension maxima admisible /6e4/
        D0     desplazamiento maximo admisible del pivote 3 /1e-1/
        h      altura de la estructura /1/;

PARAMETER K constante;
 K= F/(2*SQRT(2)*h);

POSITIVE VARIABLES x,z;
FREE VARIABLE obj;

EQUATIONS
 W      funcion objetivo
 D      desplazamiento del pivote 3
 S1     tension en el pivote 1
 S2     tension en el pivote 2;

W.. obj =e= 2*Gamma*SQRT(SQR(x)+SQR(h))*z;
D..  K*( (SQR(h)+SQR(x))**(3/2)) * SQRT(h**4+x**4) /(E*h*SQR(x)*z) =l= D0;
S1.. K*((x+h)*SQRT(SQR(x)+SQR(h))) / (x*z) =l= S0;
S2.. K*((h-x)*SQRT(SQR(x)+SQR(h))) / (x*z)=l= S0;

MODEL estructura /ALL/;

x.lo=0.05; z.lo=0.001;
x.l=100; z.l=100;

SOLVE estructura USING nlp MINIMIZING obj;
```

La solución del problema es:

$$Z = 148.667; \quad x = 0.472; \quad z = 0.329$$

11.4.8 El ejemplo de la columna

El problema de la columna se estudió en el apartado 3.3.3. La formulación del problema es como sigue.

Minimizar

$$Z = DHxy - \left(\frac{Exy^3}{4H^3 \left(M + \frac{33}{140} DHxy \right)} \right)^{1/2}$$

sujeto a

$$\begin{array}{rcl} \frac{Mg}{xy} & \leq & S \\ \frac{Mg}{xy} & \leq & \frac{\pi^2 E y^2}{48 H^2} \\ x, y & \geq & 0 \end{array}$$

Un fichero de entrada GAMS que resuelve este problema es

```
$title PROBLEMA DE LA COLUMNA

SCALARS M  masa a soportar por la columna /100/
        H  altura de la columna /10/
        D  densidad del material /100/
        E  modulo de Young del material  /1e6/
        S  maxima tension permitida /6e4/
        G  constante gravitatoria /9.8/
        Pi constante /3.141592/;

POSITIVE VARIABLES x,y;
FREE VARIABLE z;

EQUATIONS
 W    funcion objetivo
 R1   restriccion de la tension de compresion
 R2   restriccion de la tension de pandeo;

W..  z =e= D*H*x*y-SQRT((E*x*(y**3))/(4*(H**3)*(M+(33/140)*D*H*x*y)));
R1.. (M*G)/(x*y) =l= S;
R2.. (M*G)/(x*y) =l= (SQR(Pi)*E*SQR(y))/(48*SQR(H));

MODEL columna /ALL/;

x.lo=0.001; y.lo=0.001;
x.l=10; y.lo=10;

SOLVE columna USING nlp MINIMIZING z;
```

La solución de este problema es

$$Z = 19989.8; \;\; x = 2; \;\; y = 10$$

11.4.9 El ejemplo del sistema de vigas y cuerdas

El problema de las vigas y cuerdas se estudió en el apartado 1.7, y consiste en

Maximizar

$$\sum_i x_i$$

sujeto a

$$
\begin{array}{rcl}
\sum\limits_{s\in\Psi_b} t_s & = & \sum\limits_{i\in\Omega_b} x_i + \sum\limits_{x\in\Theta_b} t_s, b\in B \\
\sum\limits_{s\in\Psi_b} dr_s t_s & = & \sum\limits_{i\in\Omega_b} xl_i x_i + \sum\limits_{x\in\Theta_b} dr_s t_s, b\in B \\
0\leq t_s & \leq & T_s, s\in S \\
0\leq xl_i & \leq & l_b, i\in\Omega_b \\
0 & \leq & x_i
\end{array}
$$

Obsérvese que la formulación anterior es análoga a la versión lineal de este ejemplo. Si se ha comprendido la implementación de la versión lineal, es fácil modificar el programa GAMS para implementar el problema con no linealidades.

Un fichero de entrada GAMS que resuelve el problema es:

```
$title PROBLEMA DE LAS VIGAS Y CUERDAS (NO LINEAL)

SET
 B    conjunto de vigas   /B1*B3/
 R    conjunto de cuerdas   /RA,RB,RC,RD,RE,RF/
 L    conjunto de cargas   /L1,L2/

UPP(B,R)
      /B1.(RA,RB)
       B2.(RC,RD)
       B3.(RE,RF)/
DOWN(B,R)
      /B1.(RC,RD)
       B2.(RF)/
 LOAD(B,L)
      /B1.L1
       B3.L2/;

PARAMETER LMAX(R) carga maxima en las cuerdas
    / (RA,RB) 300
      (RC,RD) 200
      (RE,RF) 100/;

PARAMETER dr(R) coordenadas de la cuerda R
    / RA  2
      RB 12
      RC  4
      RD 12
      RE  0
      RF 10/;

VARIABLES
  z    valor de la funcion objetivo
  x(L) carga de la viga
  T(R) tension sobre la cuerda R
  d(L) distancia de la carga L respecto al extremo izquierdo
       de la viga;
```

```
POSITIVE VARIABLE d;
T.UP(R) = LMAX(R);

EQUATIONS
 COST       funcion objetivo
 FORCES(B)  restriccion de equilibrio de fuerzas
 MOMENT(B)  restriccion de equilibrio de momentos;

COST ..         z  =E=  SUM(L, x(L)) ;
FORCES(B)..     SUM(R$UPP(B,R),T(R))=E= SUM(L$LOAD(B,L),x(L))+
                SUM(R$DOWN(B,R),T(R));
MOMENT(B)..     SUM(R$UPP(B,R),dr(R)*T(R))=E=SUM(L$LOAD(B,L),
                d(L)*x(L))+SUM(R$DOWN(B,R),dr(R)*T(R));

MODEL vigaycuerda /COST,FORCES,MOMENT/;
SOLVE vigaycuerda USING nlp MAXIMIZING z;
```

Una parte del fichero de salida GAMS es:

```
v                       LOWER     LEVEL     UPPER    MARGINAL
---- VAR Z               -INF    700.000     +INF       .
  Z           valor de la funcion objetivo

---- VAR X           carga de la viga
      LOWER     LEVEL     UPPER    MARGINAL
L1     -INF    500.000     +INF       .
L2     -INF    200.000     +INF       .

---- VAR T           tension sobre la cuerda R
      LOWER     LEVEL     UPPER    MARGINAL
RA     -INF    300.000   300.000     1.000
RB     -INF    300.000   300.000     1.000
RC     -INF     25.000   200.000       .
RD     -INF     75.000   200.000       .
RE     -INF    100.000   100.000     1.000
RF     -INF    100.000   100.000      EPS

---- VAR D           distancia de la carga L respecto al extremo izquierdo
                     de la viga
      LOWER     LEVEL     UPPER    MARGINAL
L1      .        6.400     +INF       .
L2      .        5.000     +INF       .
```

y la solución es:

$Z = 700$ que se alcanza para los valores: $x_1 = 500,\ \ x_2 = 200,\ \ d_1 = 6.4,\ \ d_2 = 5.$,

y las tensiones correspondientes de las cuerdas son:

$$T_A = 300,\ \ T_B = 300,\ \ T_C = 25,\ \ T_D = 75,\ \ T_E = 100,\ \ T_F = 100$$

11.4.10 Estimación de estado en sistemas eléctricos

El problema de la estimación de estado en sistemas eléctricos que se estudió en el apartado 3.4, y en el ejemplo 3.1, es como sigue.

Minimizar

$$Z = \sum_{i \in \Omega} \frac{1}{\sigma_i^v}(v_i - \hat{v}_i)^2 + \sum_{k \in \Omega, l \in \Omega_k} \frac{1}{\sigma_{kl}^p}(p_{kl}(\cdot) - \hat{p}_{kl})^2 + \sum_{k \in \Omega, l \in \Omega_k} \frac{1}{\sigma_{kl}^q}(q_{kl}(\cdot) - \hat{q}_{kl})^2$$

sujeto a ninguna restricción, donde $\delta_k = 0$, $k = 2$, $z_{12} = 0.15$,
$\theta_{12} = 90°$, y

$$\mathbf{V} = \begin{pmatrix} 1.07 \\ 1.01 \end{pmatrix}, \quad \mathbf{P} = \begin{pmatrix} 0.83 \\ 0.81 \end{pmatrix}, \quad \mathbf{Q} = \begin{pmatrix} 0.73 \\ 0.58 \end{pmatrix}$$

Algunos comentarios sobre el código GAMS de este ejemplo son:

- Se define una matriz simétrica especificando sólo los valores de su diagonal superior. Para asignar los valores a la parte triangular inferior se emplea la expresión condicional `$(ORD(N) GT ORD(NP))` en la sentencia de asignación.

- También, se emplean expresiones condicionales para restringir el ámbito de aplicación de las restricciones. La condición `$(LINE(N,NP,'Z') NE 0)` en una restricción indica que si la impedancia Z de una línea es cero entonces esa línea no está afectada por esa restricción.

Un fichero de entrada GAMS que resuelve este problema es:

```
$title ESTIMACION DE ESTADO EN SISTEMAS ELECTRICOS

**  En primer lugar se define el conjunto de nudos.

SET
    N       indice de nudos       /N1*N2/

**  Se asignan los datos de entrada. La primera tabla representa los datos
**  para cada linea. Los identificadores de las columnas no se han
**  declarado como elementos que pertenecen a algun conjunto,
**  por eso se utiliza el simbolo '*' en la definicion de la tabla.
**  Igualmente ocurre en la segunda tabla.

TABLE LINE(N,N,*) datos de entrada de las lineas
               Z       PHI
*            (Ohm)   (grados)
    N1.N2    0.15      90;

TABLE BUS(N,*) datos de entrada de los nudos
            V       P      Q
*          (V)     (W)   (VAr)
    N1    1.07    0.83   0.73
    N2    1.01    0.81   0.58;

**  La constante PI se usa para convertir grados a radianes.

SCALAR
    PI  /3.1416/;

**  Se declaran las variables de optimizacion.
```

```
VARIABLES
    z    valor de la funcion objetivo
    V(N) modulo de la tension en el nudo N
    d(N) angulo de la tension en el nudo N;

**  El nudo 2 se toma como nudo de referencia, por
**  eso se fija su angulo a cero
    D.FX('N2')=0;

**  El conjunto de nudos N se duplica para hacer referencias
**  a distintos elementos del mismo dentro de la misma restriccion.
    ALIAS(N,NP);

**  A continuacion se convierten grados a radianes.
    LINE(N,NP,'PHI')=LINE(N,NP,'PHI')*PI/180;

**  Las matrices de datos Z y PHI se definen como simetricas mediante
**  la condicion $(ORD(N) GT ORD(NP)) sobre los conjuntos N y NP.
    LINE(N,NP,'Z')$(ORD(N) GT ORD(NP))=LINE(NP,N,'Z');
    LINE(N,NP,'PHI')$(ORD(N) GT ORD(NP))=LINE(NP,N,'PHI');

EQUATION
    ERROR    funcion objetivo;

**  La funcion objetivo se define en terminos de funciones no lineales
**  como SQR, COS y SIN. La condicion $(LINE(N,NP,'Z') NE 0) sirve
**  para comprobar si los nudos N y NP estan conectados.

ERROR..  z =e= SUM(N, SQR(V(N)-BUS(N,'V')) +SUM(NP$(LINE(N,NP,'Z') NE 0),
               SQR(((1/LINE(N,NP,'Z'))*
              (SQR(V(N))*COS(LINE(N,NP,'PHI'))-V(N)*V(NP)*COS(d(N)-
               d(NP)+LINE(N,NP,'PHI'))))-BUS(N,'P'))+
               SQR(((1/LINE(N,NP,'Z'))*
              (SQR(V(N))*SIN(LINE(N,NP,'PHI'))-V(N)*V(NP)*SIN(d(N)-
               d(NP)+LINE(N,NP,'PHI'))))-BUS(N,'Q'))));

**  Los comandos siguientes definen el modelo de estimacion, y solicitan a GAMS
**  que resuelva el problema mediante un optimizador no lineal.

MODEL estimacion /ERROR/;
SOLVE estimacion USING nlp MINIMIZING z;
```

Una parte del fichero de salida GAMS, donde se muestra la solución óptima del problema, es:

```
                       LOWER     LEVEL     UPPER    MARGINAL
---- VAR Z              -INF      2.203     +INF       .
  Z           valor de la funcion objetivo

---- VAR V           modulo de la tension en el nudo N
      LOWER     LEVEL     UPPER    MARGINAL
N1     -INF      1.045     +INF       EPS
N2     -INF      1.033     +INF       EPS

---- VAR D           angulo de la tension en el nudo N
      LOWER     LEVEL     UPPER    MARGINAL
```

```
N1     -INF      0.002     +INF       EPS
N2       .         .         .        EPS
```

y la solución es:

$$Z = 2.203$$

en los puntos:

$$v_1 = 1.045, \;\; v_2 = 1.033, \;\; \delta_1 = 0.002$$

11.4.11 Reparto óptimo de cargas

El problema del reparto óptimo de cargas que se estudió en el apartado 3.4.2, y en el ejemplo 3.2, se formula como sigue.

Minimizar

$$Z = \sum_{i=1}^{n} C_i p_{Gi}$$

sujeto a

$$\begin{array}{rcll}
p_{Gi} - P_{Di} & = & v_i \sum_{k=1}^{n} Y_{ik} v_k \cos(\delta_i - \delta_k - \Theta_{ik}), & \forall i = 1, 2, \ldots, n, \\
q_{Gi} - Q_{Di} & = & v_i \sum_{k=1}^{n} Y_{ik} v_k \sin(\delta_i - \delta_k - \Theta_{ik}), & \forall i = 1, 2, \ldots, n, \\
\underline{V}_i \leq v_i & \leq & \overline{V}_i, & \forall i = 1, 2, \ldots, n, \\
\underline{P}_{Gi} & \leq & p_{Gi} \leq \overline{P}_{Gi}, & \forall i = 1, 2, \ldots, n, \\
\underline{Q}_{Gi} & \leq & q_{Gi} \leq \overline{Q}_{Gi}, & \forall i = 1, 2, \ldots, n, \\
-\pi \leq \delta_i & \leq & \pi, & \forall i = 1, 2, \ldots, n, \\
\delta_k & = & 0. &
\end{array}$$

donde $n = 3$, y

$$\begin{array}{rcl}
\underline{P} & = & \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \;\; \overline{P} = \begin{pmatrix} 3 \\ 3 \end{pmatrix}, \;\; \underline{Q} = \begin{pmatrix} -1 \\ -1 \end{pmatrix}, \;\; \overline{Q} = \begin{pmatrix} 2 \\ 2 \end{pmatrix}, \;\; \mathbf{C} = \begin{pmatrix} 6 \\ 7 \end{pmatrix} \\
\underline{V} & = & \begin{pmatrix} 0.95 \\ 0.95 \\ 0.95 \end{pmatrix}, \;\; \overline{V} = \begin{pmatrix} 1.13 \\ 1.10 \\ 1.10 \end{pmatrix}, \;\; \mathbf{Y} = \begin{pmatrix} 22.97 & 12.13 & 10.85 \\ 12.13 & 21.93 & 9.81 \\ 10.85 & 9.81 & 20.65 \end{pmatrix} \\
\mathbf{P} & = & \begin{pmatrix} 0 \\ 0 \\ 4.5 \end{pmatrix}, \;\; \mathbf{Q} = \begin{pmatrix} 0 \\ 0 \\ 1.5 \end{pmatrix}, \;\; \mathbf{\Theta} = \begin{pmatrix} -1.338 & 1.816 & 1.789 \\ 1.816 & -1.347 & 1.768 \\ 1.789 & 1.768 & -1.362 \end{pmatrix}
\end{array}$$

Los comentarios sobre el código GAMS del problema de la estimación de estado son aplicables al siguiente código que resuelve el ejemplo de reparto de cargas óptimo:

```
$title PROBLEMA DEL FLUJO OPTIMO DE CARGAS

**  Se definen los conjuntos G y N.
```

```
**  Luego se define el conjunto MAP
**  como un subconjunto de los conjuntos G y N.
**  El subconjunto MAP establece las combinaciones
**  validas entre los elementos de G y N.

SETS
  G   indice de generadores  /G1*G2/
  N   indice de nudos /N1*N3/
  MAP(G,N) asocia generadores y nudos /G1.N1,G2.N2/;

**  Se asignan los datos de entrada. La primera tabla representa los datos
**  para cada generador. Los identificadores de las columnas no se han
**  declarado como elementos que pertenecen a algun conjunto,
**  por eso se utiliza el simbolo '*' en la definicion de la tabla.
**  Igualmente ocurre en los de las dos tablas posteriores.

TABLE GDATA(G,*)   datos de entrada de los generadores
           PMIN     PMAX     QMIN     QMAX    COST
*          (W)      (W)      (VAr)    (VAr)  (E/Wh)
    G1     0.0      3.0      -1.0     2.0      6
    G2     0.0      3.0      -1.0     2.0      7;

TABLE LINE(N,N,*) datos de entrada de las lineas
              Y       PHI
*           (Ohm)    (rad)
    N1.N1   22.97   -1.338
    N2.N2   21.93   -1.347
    N3.N3   20.65   -1.362
    N1.N2   12.13    1.816
    N1.N3   10.85    1.789
    N2.N3    9.81    1.768;

TABLE BUS(N,*)
          VMIN    VMAX   PL    QL
*         (V)     (V)    (W)  (VAr)
    N1    0.95    1.13
    N2    0.95    1.10
    N3    0.95    1.10   4.5   1.5;

**  Constante PI util para limitar los angulos de tension.
SCALAR
    PI  /3.1416/;

**  Se declaran las variables de optimizacion.
VARIABLES
    z    valor de la funcion objetivo
    p(G) potencia activa generada por el generador G
    q(G) potencia reactiva generada por el generador G
    v(N) modulo de la tension en el nudo N
    d(N) angulo de la tension en el nudo N;

**  Se asignan los limites de las variables.
    p.lo(G)=GDATA(G,'PMIN');    p.up(G)=GDATA(G,'PMAX');
    q.lo(G)=GDATA(G,'QMIN');    q.up(G)=GDATA(G,'QMAX');
    v.lo(N)=BUS(N,'VMIN');      v.up(N)=BUS(N,'VMAX');
    d.lo(N)=-PI;                d.up(N)=PI;
```

```
**  El nudo 3 se toma como nudo de referencia, por
**  eso se fija su angulo a cero.
    d.fx('N3')=0;

**  El conjunto de nudos N se duplica para hacer referencia
**  a distintos elementos del mismo dentro de la misma restriccion.
    ALIAS(N,NP);

**  Las matrices de datos Z y PHI se definen simetricas mediante
**  la condicion $(ORD(N) GT ORD(NP)) sobre los conjuntos N y NP.
    LINE(N,NP,'Y')$(ORD(N) GT ORD(NP))=LINE(NP,N,'Y');
    LINE(N,NP,'PHI')$(ORD(N) GT ORD(NP))=LINE(NP,N,'PHI');

** Se declaran las restricciones.
EQUATIONS
    COST       funcion objetivo
    PBAL(N)    restricciones de balance de potencia activa
    QBAL(N)    restricciones de balance de potencia reactiva;

** En las restricciones se emplea la condicion $MAP(G,N)
** para comprobar si el generador G esta ubicado en el nudo N.

    COST..       z =e= SUM(G,GDATA(G,'COST')*p(G));
    PBAL(N)..    SUM(G$MAP(G,N),p(G))-BUS(N,'PL')=e=v(N)*
                 SUM(NP,LINE(N,NP,'Y')*v(NP)*COS(d(N)-d(NP)
                 -LINE(N,NP,'PHI')));
    QBAL(N)..    SUM(G$MAP(G,N),q(G))-BUS(N,'QL')=e= v(N)*
                 SUM(NP,LINE(N,NP,'Y')*v(NP)*SIN(d(N)-d(NP)-LINE(N,NP,'PHI')));

**  Los comandos siguientes definen el modelo de flujo óptimo de cargas,
**  y solicitan a GAMS que resuelva el problema mediante un optimizador no lineal.

MODEL foc /COST,PBAL,QBAL/;

SOLVE foc USING nlp MINIMIZING z;
```

La parte del fichero de salida GAMS, que muestra la solución óptima es:

```
                      LOWER     LEVEL     UPPER    MARGINAL
---- VAR Z            -INF     30.312     +INF        .
  Z           valor de la funcion objetivo

---- VAR P    potencia activa generada por el generador G
      LOWER     LEVEL     UPPER    MARGINAL
G1      .        3.000     3.000    -0.927
G2      .        1.759     3.000      .

---- VAR Q    potencia reactiva generada por el generador G
      LOWER     LEVEL     UPPER    MARGINAL
G1    -1.000     1.860     2.000      .
G2    -1.000     0.746     2.000      .

---- VAR V    modulo de la tension en el nudo N
      LOWER     LEVEL     UPPER    MARGINAL
N1     0.950     1.130     1.130    -0.533
N2     0.950     1.100     1.100    -3.303
N3     0.950     0.979     1.100      .
```

```
---- VAR D    angulo de la tension en el nudo N
      LOWER     LEVEL     UPPER    MARGINAL
N1    -3.142     0.190     3.142      .
N2    -3.142     0.174     3.142      .
N3      .         .         .        EPS
```

y la solución es

$$
\begin{array}{rcl}
Z & = & 30.312 \\
p_{G1} & = & 3.000 \\
p_{G2} & = & 1.759 \\
q_{G1} & = & 1.860 \\
q_{G2} & = & 0.746 \\
v_1 & = & 1.130 \\
v_2 & = & 1.100 \\
v_3 & = & 0.979 \\
\delta_1 & = & 0.190 \\
\delta_2 & = & 0.174
\end{array}
$$

11.4.12 El problema de la red de abastecimiento de agua

En la versión lineal de este ejemplo, la función de costes se formuló como:

$$Z = \sum_{ij} c_{ij}(x_{ij}^{+} + x_{ij}^{-})$$

que se corresponde con

$$Z = \sum_{ij} c_{ij}|x_{ij}|$$

La última función, que es no lineal, se emplea en este ejemplo. Dado que esta función no es diferenciable, es necesario un optimizador especifico para funciones no lineales y no diferenciables. En el código se hace referencia a este optimizador mediante la palabra reservada `dnlp`.

Un fichero de entrada GAMS, para resolver el ejemplo cuya función objetivo es no lineal y no diferenciable es:

```
$title RED DE ABASTECIMIENTO DE AGUA (no lineal)

**  En primer lugar se declaran los conjuntos:
**  El conjunto I tiene 4 elementos.
**  CONEX se define como un subconjunto de I.
**  El subconjunto CONEX define las conexiones validas entre los I nudos.

SET
 I          conjunto de nudos en la red /I1*I4/
 CONEX(I,I) conjunto de conexiones entre nudos /I1.I2,I1.I3,I1.I4,I2.I4,I3.I4/;

**  El conjunto de nudos I se duplica para hacer referencias
**  a distintos elementos del mismo dentro de la misma restriccion.

 ALIAS(I,J)
```

```
**  Los vectores para almacenar los datos de entrada se declaran como parametros.
**  Se asignan los datos a estos vectores.
**  FMAX(I,J) es una matriz de datos, declarada como un parametro,
**  cuyos elementos tienen el valor 8. Esta es una forma compacta
**  de declarar una matriz (no es necesario usar el comando TABLE).

PARAMETERS
 F(I)   flujo de entrada y salida en el nudo I
      /I1  20
       I2  -3
       I3  -10
       I4  -7/
FMAX(I,J) capacidad maxima de la conexion entre I y J;
FMAX(I,J)=8;

**  Se declaran las variables de optimizacion.

VARIABLES
  z      valor de la funcion objetivo
  x(I,J) flujo entre el nudo I y el nudo J

**  El flujo entre I y J se representa mediante una variable positiva.
POSITIVE VARIABLE x(I,J);

**  Se limitan las variables de optimizacion.
x.lo(I,J)=-FMAX(I,J);
x.up(I,J)=FMAX(I,J);

** Se declaran las restricciones
EQUATIONS
 COST        funcion objetivo
 BALANCE(I)  restricciones de conservacion del flujo;

**  La funcion objetivo es la suma de los valores absolutos de los flujos
**  entre los nudos. Para que solo se sumen los nudos conectados
**  se utiliza la condicion $CONEX(I,J).
**  Las 4 ecuaciones de BALANCE solo consideran el flujo entre
**  nudos conectados y por eso tambien se incluye $CONEX(I,J).

COST ..         z=E=SUM((I,J)$CONEX(I,J),ABS(x(I,J)));
BALANCE(I) ..   SUM(J$CONEX(I,J),x(I,J))-SUM(J$CONEX(J,I),x(J,I))=E=F(I);

**  Los comandos siguientes declaran y definen el modelo,
**  incluyendo todas las restricciones declaradas, y solicitan
**  a GAMS que resuelva el problema mediante un optimizador para
**  funciones no lineales y no diferenciables.

MODEL raa_dnlp /ALL/;
SOLVE raa_dnlp USING dnlp MINIMIZING z;
```

Una parte del fichero de salida GAMS es:

```
                       LOWER     LEVEL     UPPER    MARGINAL
---- VAR Z              -INF     23.000     +INF       .
  Z           valor de la funcion objetivo

---- VAR X           flujo entre el nudo I y el nudo J
```

```
        LOWER     LEVEL     UPPER    MARGINAL
I1.I2   -8.000     4.000    8.000      .
I1.I3   -8.000     8.000    8.000    -2.000
I1.I4   -8.000     8.000    8.000    -1.000
I2.I4   -8.000     1.000    8.000      .
I3.I4   -8.000    -2.000    8.000      .
```

y la solución es:

$$Z = 23.000, \quad \mathbf{X} = \begin{pmatrix} 0 & 4 & 8 & 8 \\ -4 & 0 & 0 & 1 \\ -8 & 0 & 0 & -2 \\ -8 & -1 & 2 & 0 \end{pmatrix}$$

11.4.13 El problema de la matriz equilibrada

Este ejemplo se estudió en el apartado 3.5. En este problema se minimiza la siguiente función ponderada de mínimos cuadrados:

$$\sum_{i=1}^{m}\sum_{j=1}^{n}\omega_{ij}(T_{ij}-t_{ij})^2$$

sujeto a

$$\begin{array}{rcll} \sum_{j=1}^{n} t_{ij} & = & r_i & i=1,\ldots,m \\ \sum_{i=1}^{m} t_{ij} & = & c_j & j=1,\ldots,n \\ T_{ij} & \geq & 0 & i=1,\ldots,m, \;\; j=1,\ldots,n \end{array}$$

donde $m = n = 4$, y se supone que $\mathbf{r} = (12,000, 10,500, 3800, 7700)^T$, $\mathbf{c} = (6750, 7300, 10,000, 9950)^T$, $\omega_{ij} = 1/T_{ij}$ y

$$\mathbf{T} = \begin{pmatrix} - & 60 & 275 & 571 \\ 50 & - & 410 & 443 \\ 123 & 61 & - & 47 \\ 205 & 265 & 45 & - \end{pmatrix}$$

Este ejemplo no utiliza ninguna estructura nueva de GAMS. Para comprender el código se han insertado comentarios aclaratorios.

```
$Title EL PROBLEMA DE LA MATRIZ EQUILIBRADA

** En primer lugar se definen los conjuntos de indices.
** El conjunto I hace referencia a las filas de la matriz actualizada.
** El conjunto J hace referencia a las columnas de la matriz actualizada.

SET
 I indice asociado a las filas de la matriz  /R1*R4/
 J indice asociado a las columnas de la matriz /C1*C4/;

PARAMETERS
```

```
 R(I) numero de viajes estimados que parten de I
  /R1 12000
   R2 10500
   R3  3800
   R4  7700/
 C(J) numero de viajes estimados que llegan a J
  /C1  6750
   C2  7300
   C3 10000
   C4  9950/ ;

** La siguiente tabla define la matriz de datos desactualizada.

TABLE OT(I,J) matriz origen-destino resultado de una encuesta
          C1     C2      C3      C4
 R1              60     275     571
 R2       50            410     443
 R3      123     61              47
 R4      205    265      45       ;

VARIABLES
 z      valor de la funcion objetivo
 t(I,J) numero de viajes predichos desde I hasta J;

POSITIVE VARIABLES t;

** Las restricciones del problema se declaran.
** Primero, la funcion objetivo (COST).
** Esta funcion es una funcion ponderada de minimos cuadrados
** y su valor es la distancia entre los valores estimados y observados
** de las matrices de viaje. Las demas restricciones aseguran
** que la matriz estimada sume un valor determinado por filas y columnas.


EQUATIONS
 COST    funcion objetivo
 SO(I)   restricciones sobre los origenes
 SD(J)   restricciones sobre los destinos;

COST..  z=E= SUM((I,J)$(ORD(I) NE ORD(J)),POWER(t(I,J)-OT(I,J),2)/OT(I,J));
SO(I).. SUM(J$(ORD(I) NE ORD(J)),t(I,J)) =e= R(I);
SD(J).. SUM(I$(ORD(I) NE ORD(J)),t(I,J)) =e= C(J);

**  Los comandos siguientes declaran y definen el modelo,
**  incluyendo todas las restricciones declaradas, y solicitan
**  a GAMS que resuelva el problema mediante un optimizador no lineal.

MODEL pmb /ALL/;
SOLVE pmb USING nlp MINIMIZING z;

** El comando siguiente presenta la solucion en formato de tabla.
DISPLAY  t.l;
```

La parte del fichero de salida GAMS que muestra la solución óptima del problema de la matriz balance es:

```
                       LOWER     LEVEL     UPPER    MARGINAL
```

```
---- VAR Z              -INF  441754.6224   +INF       .
  Z           valor de la funcion objetivo

----     73 VARIABLE  T.L           numero de viajes predichos desde I hasta J

            C1          C2          C3          C4

R1                  1670.878    4286.405    6042.717
R2    1226.653                  5522.850    3750.497
R3    2386.544    1256.670                   156.786
R4    3136.803    4372.452     190.745
```

La solución del problema se muestra en el apartado 3.5.

11.4.14 El problema de la asignación de tráfico

El problema de la asignación de tráfico se estudió en el apartado 3.6. La formulación de este problema es como sigue.

Minimizar

$$Z = \sum_{a \in \mathcal{A}} c^0_{a_i} f_{a_i} + \frac{b_{a_i}}{n_{a_i}+1} \left(\frac{f_{a_i}}{k_{a_i}} \right)^{n_{a_i}+1}$$

sujeto a

$$\begin{array}{lcl} h_1 + h_2 + h_3 & = & 4000 \\ h_4 + h_5 & = & 2500 \\ h_1 & = & f_1 \\ h_2 + h_4 & = & f_2 \\ h_3 + h_5 & = & f_3 \\ h_2 + h_3 & = & f_4 \\ h_1, \dots, h_5 & \geq & 0 \end{array}$$

Un fichero de entrada GAMS para resolver este problema es:

```
$title EL PROBLEMA DE ASIGNACION DE TRAFICO

SETS
r rutas         /r1*r5/
a arcos         /a1*a4/
w demandas      /w1*w2/;

PARAMETERS
D(w) demanda
/w1 4000
 w2 2500/
K(a) capacidad del arco
 /a1 500
  a2 400
  a3 400
  a4 500/
b(a) parametro de congestion del arco
 /a1   1.
  a2   1.
  a3   1.
  a4   1./
n(a) parametro de congestion del arco
```

```
 /a1   4.
  a2   4.
  a3   4.
  a4   4./
C0(a) coste del arco sin flujo
  / a1   5
    a2   7
    a3  10
    a4   2/ ;
TABLE DELTAR_W(r,w) matriz de incidencias ruta-demanda
         w1      w2
r1       1       0
r2       1       0
r3       1       0
r4       0       1
r5       0       1
TABLE  DELTAR_L(r,a) matriz de incidencias ruta-arco
         a1      a2      a3      a4
r1       1       0       0       0
r2       0       1       0       1
r3       0       0       1       1
r4       0       1       0       0
r5       0       0       1       0;

VARIABLES
H(r) flujo en la ruta r
F(a) flujo en el arco a
Z    tiempo total en la red de transporte
POSITIVE VARIABLE H, F;

EQUATIONS
COST    funcion objetivo
SD(w)   restriccion sobre la demanda
CF(a)   conservacion del flujo;

SD(w).. SUM(r,H(r)*DELTAR_W(r,w))=e= D(w);
CF(a).. SUM(r,H(r)*DELTAR_L(r,a))=e= F(a);
COST..Z=e=SUM(a,C0(a)*F(a)+(b(a)/(n(a)+1.))*(F(a)/K(a))**(n(a)+1.) );

MODEL PAT /all/;
SOLVE PAT using nlp minimizing Z;
DISPLAY H.L;
DISPLAY F.L;
```

La solución del problema se presenta al final del ejemplo 3.4.

Ejercicios

11.1 Se quiere realizar un estudio de transporte entre tres ciudades A, B, y C. El coste de viaje entre ciudades se da en la tabla 11.1. Obsérvese que no se consideran viajes intraurbanos en este estudio.

Se han realizado encuestas en todas las ciudades. El número de conductores encuestados por ciudad se muestra en la tabla 11.2, donde se indican sus orígenes y destinos asociados.

Tabla 11.1: Coste de transporte (en dolares) entre las ciudades

	Destinos		
Orígenes	A	B	C
A	–	1.80	2.60
B	1.80	–	1.52
C	2.70	1.52	–

Tabla 11.2: Resultados de la encuesta de transporte

	Destinos		
Orígenes	A	B	C
A	–	6	2
B	8	–	8
C	6	6	–

Supóngase que el modelo de transporte se puede expresar mediante la siguiente función

$$T_{ij} = a_i b_j \exp(-\beta c_{ij})$$

donde a_i es una constante que depende del origen i, b_j depende del destino j, β es un parámetro, y c_{ij} es el coste del viaje entre ciudades. Formúlese un problema de estimación de mínimos cuadrados para ajustar los parámetros del modelo. Obténgase la solución mediante la herramienta GAMS.

11.2 Walter fabrica dos tipos de transformadores y tiene a su disposición 6 toneladas de material ferromagnético y 28 horas de trabajo. El transformador 1 requiere 2 toneladas de este material y el transformador 2 sólo requiere 1 tonelada. Los precios de los transformadores 1 y 2 son, respectivamente, 120 y 80 miles de euros ¿Cuántos transformadores de cada tipo deberá fabricar Walter para maximizar sus beneficios?. Resuélvase el problema gráficamente y analíticamente (mediante GAMS).

11.3 Encuéntrese la solución óptima del siguiente problema. Maximizar

$$Z = \sum_{j=1}^{100} \frac{2^j}{j^2} x_j$$

sujeto a

$$0 \leq \sum_{j=i}^{100} (-1)^{j+i} x_j \leq 1 \text{ for } i = 1, \ldots, 100$$

mediante GAMS.

11.4 Mediante la herramienta de programación GAMS, encuéntrese la solución del siguiente problema. Minimizar

$$Z = x$$

sujeto a

$$x \ln x \geq 1$$

usando los siguientes métodos:

(a) Método de penalización de barrera logarítmica

(b) Método de penalización de barrera inversa

La finalidad de este ejercicio es usar los bucles de la herramienta GAMS para ilustrar el funcionamiento de los métodos, a pesar de que ésta no sea la manera más eficiente de implementar los métodos.

11.5 El modelo de programación binivel se formula como sigue. Minimizar

$$Z_U = F(\mathbf{x}, \hat{\mathbf{y}})$$

sujeto a

$$\mathbf{x} \in X$$

donde $\hat{\mathbf{y}}$ es óptimo para minimizar

$$Z_L = f(\mathbf{y}, \mathbf{x})$$

sujeto a

$$\mathbf{y} \in Y(\mathbf{x})$$

donde $F : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ y $f : \mathbb{R}^m \times \mathbb{R}^n \to \mathbb{R}$ son funciones conocidas, X es un subconjunto dado, e $Y(\mathbf{x})$ es un conjunto parametrizado por la variable $\mathbf{x}$.

El objetivo $\min_{\mathbf{x} \in X} F(\mathbf{x}, \mathbf{y})$ se denomina *problema del nivel superior*, y el *problema del nivel inferior* se define como $\min_{\mathbf{y} \in Y(\mathbf{x})} f(\mathbf{y}, \mathbf{x})$ para un $\mathbf{x}$ fijo.

Se emplea el siguiente algoritmo heurístico de 4 pasos [(a)–(d)] para aproximar la solución de problemas de gran dimensión.

(a) Sea $\mathbf{x}^{(1)}$ un punto factible de X. Sea T un valor entero positivo. Hacer $t = 1$ e ir al paso (b).

(b) Se resuelve el problema del nivel inferior. Minimizar

$$Z_L = f(\mathbf{y}, \mathbf{x}^{(\mathbf{t})})$$

sujeto a

$$\mathbf{y} \in Y(\mathbf{x}^{(t)})$$

y su solución se denota mediante $\mathbf{y}^{(t)}$.

(c) Se resuelve el problema del nivel superior. Minimizar

$$Z_U = F(\mathbf{x}, \mathbf{y}^{(t)})$$

sujeto a

$$\mathbf{x} \in X$$

y su solución se denota mediante $\mathbf{x}^{(t+1)}$.

(d) Si $t = T$, entonces *parar*. Si no $t = t + 1$ e ir al paso (b).

Impleméntese el algoritmo anterior y aplicarlo al siguiente problema binivel. Minimizar

$$Z_U = (x - 8)^2(\hat{y} - 3)^2$$

sujeto a

$$\begin{array}{rcr} x & \geq & 0 \\ x & \leq & 16 \end{array}$$

donde $\hat{y}$ es el valor óptimo del problema que minimiza

$$Z_L = y$$

sujeto a

$$\begin{array}{rrcr} & y & \geq & 0 \\ x & +2y & \geq & 10 \\ x & -2y & \leq & 6 \\ 2x & -y & \leq & 21 \\ x & +2y & \leq & 38 \\ -x & +2y & \leq & 18 \end{array}$$

11.6 Altérese el modelo de despacho económico de forma que la función de los costes de producción de cada generador sea una función convexa lineal a tramos.

11.7 Altérese el modelo de la programación horaria de centrales térmicas, de forma que la función de los costes de producción de cada generador sea una función convexa lineal a tramos.

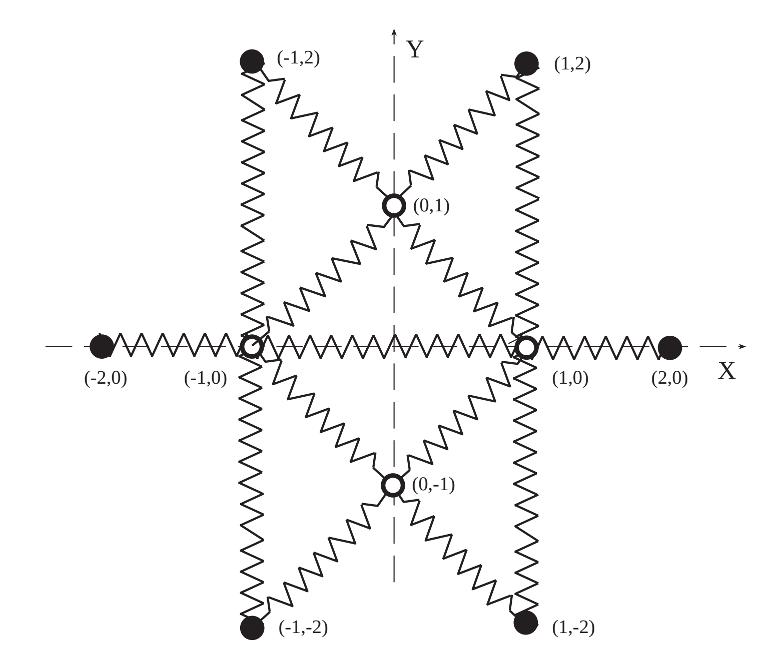


Figura 11.1: Sistema de muelles.

11.8 Altérese el modelo del reparto óptimo de cargas para que los módulos de las tensiones tomen el valor 1. Compárense los resultados obtenidos con los del modelo original.

11.9 Determinar el estado de equilibrio del sistema de la figura 11.1, con la información que proporciona el Ejercicio 3.8 del capítulo 3. Empléese la herramienta de programación GAMS.

Parte IV

Aplicaciones

Capítulo 12

Aplicaciones

En este capítulo se presentan algunas aplicaciones que ilustran cómo se combina la programación matemática con una herramienta como GAMS, que permite resolver, de manera eficiente y sencilla, distintos problemas. Este capítulo comienza con una aplicación a las redes neuronales y funcionales en la Sección 12.1. En la sección 12.2 se presenta un método de generación automática de redes que resulta de utilidad para los métodos de los elementos finitos y para todos aquellos métodos que requieren la descripción de redes por medio de elementos triangulares. En la sección 12.3 se presentan algunas aplicaciones de probabilidad. En la sección 12.4 se muestran algunas aplicaciones a problemas de optimización continuos, como la braquistocrona, la cuerda colgante, los problemas de control óptimo para alcanzar un blanco, y para controlar un oscilador armónico, y los problemas de transporte y de programación horaria de centrales térmicas. En el apartado 12.6 se presentan las aplicaciones correspondientes a sistemas de transporte urbano. Finalmente, en la sección 12.7 se muestra una aplicación a los sistemas de energía eléctrica.

12.1 Aplicaciones a la inteligencia artificial

Este apartado muestra cómo se puede aplicar la programación matemática a problemas de redes neuronales y funcionales. Esta aplicación está basada en Castillo et al. [20].

Las redes neuronales constan de una o varias capas de neuronas conectadas mediante enlaces. Cada neurona calcula un valor de salida a partir de una combinación lineal de los valores de entrada, procedentes de la capa previa, mediante una función escalar de activación conocida.

Sea la red neuronal monocapa de la figura 12.1, donde la función neuronal o función de activación f se supone invertible.

Un problema típico de redes neuronales consiste en “aprender” los valores umbral y los pesos a partir de un conjunto de datos de entrenamiento

$$D \equiv \{(x_{1s}, x_{2s}, \ldots, x_{Is}, y_s) | s = 1, 2, \ldots, S\}$$

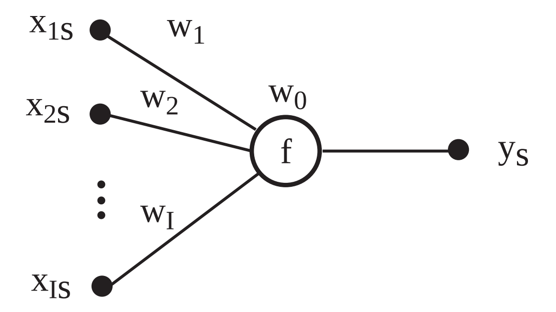


Figura 12.1: Red neuronal de una capa y su función neuronal.

que contienen un conjunto de entradas $(x_{1s}, x_{2s}, \ldots, x_{Is})$ y las salidas correspondientes y_s para una colección de S mediciones.

El conjunto de ecuaciones que expresan las salidas como una función de las entradas es

$$y_s = f(w_0 + \sum_{i=1}^{I} w_i x_{is}) + \delta_s; \quad s = 1, 2, \ldots, S \tag{12.1}$$

donde w_0 y $w_i; i = 1, 2, \ldots, I$ son el valor umbral y los pesos de las neuronas, respectivamente, y δ_s mide el error asociado a la salida y_s.

El aprendizaje de los umbrales y pesos puede modelarse mediante los siguientes problemas:

Opción 1. Minimizar

$$Q_1 = \sum_{s=1}^{S} \delta_s^2 = \sum_{s=1}^{S} \left(y_s - f(w_0 + \sum_{i=1}^{I} w_i x_{is}) \right)^2 \tag{12.2}$$

que es la suma de los cuadrados de los errores, medidos en las unidades de la salida (las unidades de y_s). Sin embargo, el anterior es un problema de programación no lineal, debido a la presencia de la función neuronal f, que es típicamente no lineal. Por tanto, no se garantiza que Q_1 tenga un óptimo global. De hecho, suelen tener múltiples óptimos locales.

Opción 2. Minimizar la suma de los cuadrados de los errores:

$$Q_2 = \sum_{s=1}^{S} \epsilon_s^2 = \sum_{s=1}^{S} \left(w_0 + \sum_{i=1}^{I} w_i x_{is} - f^{-1}(y_s) \right)^2 \tag{12.3}$$

Esta opción mide los errores en unidades de la entrada (las unidades de x_{is}). La ventaja de esta formulación es que se tiene un problema estándar de optimización de mínimos cuadrados, que puede plantearse como un sistema de

ecuaciones lineales (problema lineal).

Opción 3. Alternativamente, se puede minimizar el error máximo:

$$\min_{w} \left\{ \max_{s} \left| w_0 + \sum_{i=1}^{I} w_i x_{is} - f^{-1}(y_s) \right| \right\} \tag{12.4}$$

que se puede expresar como el problema de programación lineal. Minimizar ϵ sujeto a

$$\left.\begin{array}{rcl} w_0 + \sum_{i=1}^{I} w_i x_{is} - \epsilon & \leq & f^{-1}(y_s) \\ -w_0 - \sum_{i=1}^{I} w_i x_{is} - \epsilon & \leq & -f^{-1}(y_s) \end{array}\right\}; \quad s = 1, 2, \ldots, S \tag{12.5}$$

cuyo óptimo global se puede obtener fácilmente empleando técnicas de programación lineal

12.1.1 Aprendizaje de funciones neuronales

Un problema interesante consiste en aprender la función f^{-1} en lugar de suponerla conocida. Cuando se aprende la función de activación se tiene una red funcional. De manera más precisa, se puede suponer que f^{-1} es una combinación lineal convexa del conjunto $\{\phi_1(x), \phi_2(x), , \ldots, \phi_R(x)\}$ de las siguientes funciones básicas invertibles:

$$f^{-1}(x) = \sum_{r=1}^{R} \alpha_r \phi_r(x) \tag{12.6}$$

donde $\{\alpha_r;\ r = 1, 2, \ldots, R\}$ es el conjunto correspondiente de coeficientes, elegidos de forma que la función f^{-1} sea invertible. Sin pérdida de generalidad, se puede suponer que son crecientes.

Por tanto, se pueden considerar dos nuevas opciones:

Opción 4. Minimizar, respecto a $w_i; i = 0, 1, \ldots, I$ y $\alpha_r; r = 1, 2, \ldots, R$, la función

$$Q_4 = \sum_{s=1}^{S} \epsilon_s^2 \quad = \quad \sum_{s=1}^{S} \left(w_0 + \sum_{i=1}^{I} w_i x_{is} - \sum_{r=1}^{R} \alpha_r \phi_r(y_s) \right)^2 \tag{12.7}$$

sujeto a

$$\sum_{r=1}^{R} \alpha_r \phi_r(y_{s_1}) \quad \leq \quad \sum_{r=1}^{R} \alpha_r \phi_r(y_{s_2}); \ \ \forall y_{s_1} < y_{s_2} \tag{12.8}$$

que obliga a que la función f^{-1} sea creciente en el intervalo correspondiente, y

$$\sum_{r=1}^{R} \alpha_r \phi_r(y_0) \quad = \quad 1 \tag{12.9}$$

que evita que la función nula se tome como función óptima.

Opción 5. Alternativamente, se puede minimizar el error máximo:

$$Q_5 = \min_{w,\alpha} \left\{ \max_s \left| w_0 + \sum_{i=1}^{I} w_i x_{is} - \sum_{r=1}^{R} \alpha_r \phi_r(y_s) \right| \right\} \quad (12.10)$$

que puede expresarse mediante el siguiente problema de programación lineal. Minimizar ϵ sujeto a

$$\begin{array}{rcl} w_0 + \sum\limits_{i=1}^{I} w_i x_{is} - \sum\limits_{r=1}^{R} \alpha_r \phi_r(y_s) - \epsilon & \leq & 0 \\ -w_0 - \sum\limits_{i=1}^{I} w_i x_{is} + \sum\limits_{r=1}^{R} \alpha_r \phi_r(y_s) - \epsilon & \leq & 0 \\ \sum\limits_{r=1}^{R} \alpha_r \phi_r(y_{s_1}) & \leq & \sum\limits_{r=1}^{R} \alpha_r \phi_r(y_{s_2});\ \forall y_{s_1} < y_{s_2} \\ \sum\limits_{r=1}^{R} \alpha_r \phi_r(y_0) & = & 1 \end{array} \quad (12.11)$$

cuyo óptimo global se puede obtener sin dificultad.

Para usar la red, es decir, para calcular los valores de salida conocidos los de entrada, es necesario conocer la función $f(x)$. Sin embargo, el método anterior sólo puede obtener la función f^{-1}. Un procedimiento eficiente que permite obtener $f(x)$ cuando se conoce f^{-1} es el método de la bisección.

Los cuatro elementos principales del problema de las redes neuronales y funcionales son

1. **Datos.**

   S: número de vectores de datos
   I: dimensión de los vectores de datos de entrada
   x_{is}: vectores de datos de entrada $(x_{1s}, x_{2s}, \ldots, x_{Is}); s = 1, 2, \ldots, S$
   y_s: datos de salida $y_s; s = 1, 2, \ldots, S$

2. **Variables.**

   w_0: valor umbral de la neurona
   w_i: peso de la neurona para el elemento i del vector de entradas
   ϵ: error máximo (esta es una variable no negativa que sólo tiene sentido para la opción 3)

3. **Restricciones.** Las opciones 1 y 2 no tienen restricciones. Para la opción 3, las restricciones son

$$\left. \begin{array}{rcl} w_0 + \sum\limits_{i=1}^{I} w_i x_{is} - \epsilon & \leq & f^{-1}(y_s) \\ -w_0 - \sum\limits_{i=1}^{I} w_i x_{is} - \epsilon & \leq & -f^{-1}(y_s) \end{array} \right\};\ s = 1, \ldots, S \quad (12.12)$$

En el caso de la opción 4, se tiene

$$\begin{aligned}\sum_{r=1}^{R} \alpha_r \phi_r(y_{s_1}) &\leq \sum_{r=1}^{R} \alpha_r \phi_r(y_{s_2}); \;\; \forall y_{s_1} < y_{s_2} \\ \sum_{r=1}^{R} \alpha_r \phi_r(y_0) &= 1\end{aligned}$$

y para la opción 5, a las dos últimas restricciones se deben añadir las siguientes

$$\begin{aligned}w_0 + \sum_{i=1}^{I} w_i x_{is} - \sum_{r=1}^{R} \alpha_r \phi_r(y_s) - \epsilon &\leq 0 \\ -w_0 - \sum_{i=1}^{I} w_i x_{is} + \sum_{r=1}^{R} \alpha_r \phi_r(y_s) - \epsilon &\leq 0\end{aligned}$$

4. **Función a optimizar.** Se tienen las siguientes posibilidades:

Opción 1. Minimizar

$$Q_1 = \sum_{s=1}^{S} \delta_s^2 = \sum_{s=1}^{S} \left(y_s - f(w_0 + \sum_{i=1}^{I} w_i x_{is}) \right)^2 \tag{12.13}$$

Opción 2. Minimizar

$$Q_2 = \sum_{s=1}^{S} \epsilon_s^2 = \sum_{s=1}^{S} \left(w_0 + \sum_{i=1}^{I} w_i x_{is} - f^{-1}(y_s) \right)^2 \tag{12.14}$$

Opción 3 y 5. Minimizar ϵ.

Opción 2. Minimizar

$$Q_4 = \sum_{s=1}^{S} \epsilon_s^2 = \sum_{s=1}^{S} \left(w_0 + \sum_{i=1}^{I} w_i x_{is} - \sum_{r=1}^{R} \alpha_r \phi_r(y_s) \right)^2$$

Una vez identificados los cuatro elementos principales, se pasa a describir el problema.

Ejemplo 12.1 (red neuronal). Considérense los datos de las tres primeras columnas de la tabla 12.1, que representan las salidas y_s obtenidas para los valores de entrada $\{x_{1s}, x_{2s}\}$ con $s = 1, \ldots, 30$. Los datos se han simulado tomando el modelo

$$y_s = (0.3 + 0.3x_{1s} + 0.7x_{2s})^2 + \epsilon_s$$

donde $x_{is} \sim U(0, 0.5)$ y $\epsilon_s \sim U(-0.005, 0.005)$, es decir, distribuciones uniformes.

Con el objeto de aprender la relación $y_s = q(x_1, x_2)$, a partir de estos datos, se decide emplear una red neuronal monocapa con dos entradas $\{x_1, x_2\}$ y una salida y, y emplear las tres opciones en que se conoce la función de activación, y las dos opciones en las que esta función se aprende. Para ilustrar el proceso de aprendizaje, se ha empleado la función de activación $f(x) = \arctan(x)$, y las siguientes funciones:

$$\{\phi_1(x), \phi_2(x), \phi_3(x)\} = \{\sqrt{x}, x, x^2\}$$

El siguiente código en GAMS resuelve este problema usando el método de la bisección. Obsérvese que se han definido cinco modelos y cinco funciones objetivo distintas.

```
$title Redes neuronales (neuronal2)

SETS
S numero de vectores de datos /1*30/
I dimension de los vectores de datos de entrada /1*2/
J numero funciones de aprendizaje distintas /1*5/
R numero de funciones basicas /1*3/;

ALIAS(S,S1)

PARAMETERS
X(I,S) vectores de datos de entrada
Y(S) datos de salida
powers(R) exponentes de x en las funciones basicas
/1  0.5
 2  1
 3  2/;

X(I,S)=uniform(0,1)*0.5;
Y(S)=sqr(0.3+0.3*X('1',S)+0.7*X('2',S))+(uniform(0,1)-0.5)*0.01;

PARAMETERS
aux Parametro auxiliar
x1 Parametro auxiliar
x2 Parametro auxiliar
x3 Parametro auxiliar
f1 Parametro auxiliar
f2 Parametro auxiliar
f3 Parametro auxiliar
ymin   Valor minimo de Y(S)
ymax   Valor maximo de Y(S)
acterror Maximo error de prediccion
maxerror Maximo error permitido para el metodo de la biseccion;

maxerror=0.00001;

VARIABLES
z1 Funcion1 a optimizar
z2 Funcion2 a optimizar
z3 Funcion3 a optimizar
z4 Funcion21 a optimizar
z5 Funcion21 a optimizar
```

```
W0 Valor umbral
W(I) Peso asociado al elemento i de la entrada
alfa(R) Coeficientes de la funcion neuronal;

POSITIVE VARIABLES
epsilon Error;

EQUATIONS
Q1 Definicion de z1
Q2 Definicion de z2
Q3 Definicion de z3
Q4 Definicion de z4
Q5 Definicion de z5
const1(S) Limite superior del error
const2(S) Limite inferior del error
const3(S) Limite superior del error
const4(S) Limite inferior del error
normalized Valores normalizados
increasing(S,S1) La funcion f debe ser creciente;

Q1..z1=e=sum(S,sqr(Y(S)-arctan(W0+sum(I,W(I)*X(I,S)))));
Q2..z2=e=sum(S,sqr(W0+sum(I,W(I)*X(I,S))-sin(Y(S))/cos(Y(S))));
Q3..z3=e=epsilon;
Q4..z4=e=sum(S,sqr(W0+sum(I,W(I)*X(I,S))
            -sum(R,alfa(R)*Y(S)**(powers(R)))));
Q5..z5=e=epsilon;
const1(S)..W0+sum(I,W(I)*X(I,S))-epsilon=l=sin(Y(S))/cos(Y(S));
const2(S)..-W0-sum(I,W(I)*X(I,S))-epsilon=l=-sin(Y(S))/cos(Y(S));
const3(S)..W0+sum(I,W(I)*X(I,S))
      -sum(R,alfa(R)*Y(S)**(powers(R)))-epsilon=l=0.0;
const4(S)..-W0-sum(I,W(I)*X(I,S))
      +sum(R,alfa(R)*Y(S)**(powers(R)))-epsilon=l=0.0;
normalized..sum(R,alfa(R)*Y('1')**(powers(R)))=e=1;
increasing(S,S1)$(Y(S)<Y(S1))..sum(R,alfa(R)*Y(S)**(powers(R)))
      =l=sum(R,alfa(R)*Y(S1)**(powers(R)));;

MODEL reg1/Q1/;
MODEL reg2/Q2/;
MODEL reg3/Q3,const1,const2/;
MODEL reg4/Q4,normalized,increasing/;
MODEL reg5/Q5,const3,const4,normalized,increasing/;

file out/neural3.out/;
put out;

put "Datos"/;
ymin=0.0;
ymax=0.0;
loop(S,
  loop(I,put X(I,S):12:3," & ";);
  put Y(S):12:3,"\\"/;
  if(ymin<Y(S),ymin=Y(S));
  if(ymax>Y(S),ymax=Y(S));
);
loop(J,
  if(ord(J) eq 1, SOLVE reg1 USING nlp MINIMIZING z1;
        put "z1=",z1.l:12:9/;);
```

```
  if(ord(J) eq 2, SOLVE reg2 USING nlp MINIMIZING z2;
        put "z2=",z2.l:12:9/;);
  if(ord(J) eq 3, SOLVE reg3 USING  lp MINIMIZING z3;
        put "z3=",z3.l:12:9/;);
  if(ord(J) eq 4, SOLVE reg4 USING nlp MINIMIZING z4;
        put "z4=",z4.l:12:9/;);
  if(ord(J) eq 5, SOLVE reg5 USING  lp MINIMIZING z5;
        put "z5=",z5.l:12:9/;);
  put "Pesos:"/;
  put W0.l:12:3;
  loop(I,put " & ",W.l(I):12:3;);
  put " "/;
  if(((ord(J) eq 4) or (ord(J) eq 5)),
    put "Alfas:"/;
    loop(R,put " & ",alfa.l(R):12:3;);
    put " "/;
  );
  acterror=-10000;
  put "Datos y valores ajustados:"//;

  loop(S,
    aux=W0.l+sum(I,W.l(I)*X(I,S));
    if(ord(J) le 3,
      f1=arctan(aux);
    else
      x1=ymin-(ymax-ymin)*0.1;
      x2=ymax+(ymax-ymin)*0.1;
      f1=sum(R,alfa.l(R)*abs(x1)**(powers(R)))-aux;
      f2=sum(R,alfa.l(R)*abs(x2)**(powers(R)))-aux;
      if(f1*f2>0.0,
        put "ERROR EN LA BISECTION "," S=",S.tl:3," aux=",
                aux:12:3," f1=",f1:12:3," f2=",f2:12:3/;
      else
        while(abs(x1-x2)>maxerror,
          x3=(x1+x2)*0.5;
          f3=sum(R,alfa.l(R)*x3**(powers(R)))-aux;
          if(f3*f1>0.0,
            x1=x3;f1=f3;
          else
            x2=x3;f2=f3;
          );
        );
      );
      f1=((x1+x2)*0.5);
    );
    loop(I,put X(I,S):12:3," & ";);
    aux=(f1-Y(S));
    if(abs(aux)>acterror,acterror=abs(aux));
    put Y(S):12:3," & ",aux:12:3"\\"/;
  );
  put "Error maximo=",acterror:12:9/;
);
```

Los errores correspondientes se muestran en la tabla 12.1. Finalmente, la tabla 12.2 muestra los valores umbral, los pesos, y los coeficientes α (solo para las opciones 4 y 5), junto con los valores óptimos para cada función objetivo.

Tabla 12.1: Datos de entrada y salida para una red neuronal monocapa

Entrada		Salida	ϵ_s Errores				
x_{1s}	x_{2s}	y_s	Q_1	Q_2	Q_3	Q_4	Q_5
0.086	0.055	0.129	-0.023	-0.024	-0.020	0.009	0.004
0.422	0.251	0.360	0.005	0.007	0.012	0.007	0.008
0.275	0.080	0.193	-0.002	-0.003	0.002	-0.004	-0.004
0.151	0.436	0.426	-0.015	-0.014	-0.006	-0.002	-0.008
0.146	0.133	0.187	0.000	-0.001	0.005	-0.001	-0.003
0.112	0.143	0.183	0.000	-0.001	0.005	0.001	-0.001
0.175	0.297	0.315	0.006	0.006	0.013	-0.002	-0.002
0.428	0.361	0.466	-0.024	-0.022	-0.016	-0.004	-0.008
0.034	0.314	0.280	0.008	0.008	0.015	-0.003	-0.005
0.250	0.232	0.287	0.010	0.010	0.016	-0.001	0.000
0.499	0.207	0.351	0.007	0.008	0.013	0.006	0.009
0.289	0.059	0.181	-0.002	-0.002	0.002	0.000	-0.001
0.496	0.157	0.308	0.013	0.014	0.018	0.005	0.009
0.381	0.023	0.190	-0.007	-0.008	-0.004	-0.007	-0.006
0.065	0.169	0.191	-0.002	-0.003	0.003	-0.003	-0.006
0.320	0.091	0.214	0.000	0.000	0.004	-0.007	-0.006
0.080	0.323	0.300	0.008	0.009	0.016	0.000	-0.002
0.125	0.280	0.281	0.011	0.011	0.018	0.000	0.000
0.334	0.385	0.451	-0.021	-0.019	-0.013	-0.004	-0.008
0.218	0.149	0.216	0.008	0.008	0.013	-0.001	-0.001
0.180	0.331	0.340	0.006	0.007	0.014	0.004	0.002
0.176	0.378	0.376	0.002	0.003	0.010	0.007	0.004
0.066	0.314	0.289	0.009	0.009	0.017	-0.001	-0.003
0.075	0.142	0.178	-0.007	-0.009	-0.003	-0.003	-0.006
0.295	0.043	0.172	-0.003	-0.004	0.000	0.002	0.001
0.415	0.051	0.209	0.007	0.007	0.010	0.000	0.001
0.115	0.321	0.311	0.008	0.008	0.015	0.000	-0.001
0.333	0.273	0.347	0.006	0.007	0.012	0.004	0.005
0.388	0.016	0.181	-0.002	-0.002	0.001	0.000	0.000
0.152	0.396	0.393	-0.009	-0.008	-0.001	-0.003	-0.007
Máximo error			0.024	0.024	0.020	0.009	0.009

Si se comparan los errores para las funciones objetivo Q_4 y Q_5, para las que se aprenden las funciones neuronales, se obtienen, como es de esperar, errores más pequeños.

■

Tabla 12.2: Valores de las funciones objetivo, valores umbral, pesos, y coeficientes asociados a las cinco opciones

Función objetivo	Umbral	Pesos		Coeficientes α		
Q	w_0	w_1	w_2	α_1	α_2	α_3
$Q_1 = 0.00292$	0.032	0.353	0.803	–	–	–
$Q_2 = 0.00375$	0.029	0.358	0.812	–	–	–
$Q_3 = 0.01978$	0.034	0.354	0.823	–	–	–
$Q_4 = 0.00086$	0.940	0.369	0.847	4.199	−4.238	2.384
$Q_5 = 0.00988$	0.924	0.415	0.912	4.200	−4.269	2.596

12.2 Aplicaciones a CAD

En esta sección se presenta una aplicación a la generación automática de redes. La generación automática de redes es un campo de interés para muchos investigadores. El procedimiento más empleado en la generación automática de redes de dos dimensiones es el método de generación de triángulos, mientras que para redes de tres dimensiones se emplea el método de generación de tetraedros y hexaedros. La mayoría de los problemas de generación de redes requieren la reconstrucción mediante elementos que particionen las superficies tridimensionales. Las principales técnicas de generación de superficies se pueden clasificar en tres grupos:

1. **Técnicas basadas en el espacio paramétrico.** Puesto que las superficies en forma paramétrica tienen una representación $u - v$, se pueden usar técnicas estándar asociadas al espacio $x - y - z$ (véase, por ejemplo, Cuilliere [29] y Farouki [36]). La principal desventaja de está técnica es que su aplicación directa puede provocar que los elementos obtenidos no tengan formas bien definidas. Por tanto, suele requerir algunas modificaciones.

2. **Técnicas directas.** Estos métodos avanzan sobre la propia superficie, creando elementos basados en los previamente generados, y emplean funciones tangentes y normales a la superficie para calcular la dirección de avance.

3. **Técnicas basadas en optimización.** Estos métodos se basan en la optimización de una función de calidad sujeta a un conjunto de restricciones. Los nudos se mueven en la dirección en que se produce una mejora en la función objetivo (véanse los trabajos de Canann et al. [19], Freitag [42] y Li [67]).

En el problema de la generación de redes, dado que la mayoría de los métodos de generación obtienen redes no óptimas, normalmente se necesita un postprocesamiento que mejore la calidad de los elementos. Las principales mejoras se pueden agrupar en dos categorias

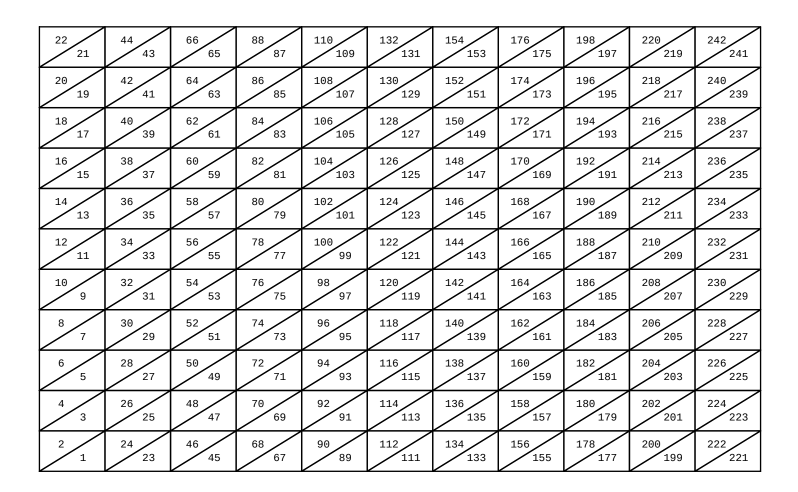


Figura 12.2: Topología de la red seleccionada.

1. **Alisar.** Consiste en ajustar la situación de los nudos sin modificar las interconexiones.
2. **Pulir.** Consiste en modificar la topología o las interconexiones entre elementos. Los métodos de pulido utilizan algunos criterios de calidad para operar, y pueden combinarse con algunas técnicas de alisado (véase Freitag [44]).
3. **Refinar.** Consiste en reducir el tamaño de los elementos. La reducción de tamaño puede ser útil para reproducir mejor la realidad localmente, o para mejorar la calidad de los elementos. De hecho, el refinamiento de una red puede considerarse como una generación de redes nuevas. Por tanto, se pueden aplicar las técnicas de generación de superfices.

12.2.1 Generación automática de mallas

En este apartado se describe cómo generar o mejorar una red usando un método de optimización. Esta aplicación se basa en Castillo y Mínguez [23].

El método parte de una topología de la red conocida en proyección (la proyección de la red de la superficie sobre el plano X–Y), como se muestra en la figura 12.2. En esta proyección, los nudos esquina se suponen fijos, los nudos frontera se pueden mover en su frontera, y los nudos interiores se pueden mover en cualquier dirección. La triangulación de la superficie debe minimizar la desviación típica o la varianza de las áreas del triángulo. Por tanto, se busca una red que presenta la mayor similitud en términos de las áreas del triángulo. En otras palabras, se minimiza

$$Q = \sum_{i=1}^{m} \frac{[a(i) - \bar{a}]^2}{m}$$

sujeto a las restricciones frontera, donde $a(i)$ es el área del triángulo i, $\bar{a}$ es la media correspondiente, y m es el número de triángulos.

Resumiendo, se tienen los siguientes pasos:

Paso 1. Se define la topología de la red mediante una red proyectada La definición compacta de una red sencilla viene dada en la tabla 12.3, donde (x_{min}, y_{min}) y (x_{max}, y_{max}) son las coordenadas de las esquinas superior izquierda y derecha, y n es el número de nudos por columna ($n = 12$ para la red de la figura 12.2). La red resultante tiene n^2 nudos y $2(n-1)^2$ elementos.

Se definen las tres componentes principales de la red como:

(a) Coordenadas de los nudos

(b) Nudos que pertenecen a cada elemento

(c) Nudos frontera

Paso 2. La superficie a triangular se define mediante las ecuaciones correspondientes.

Paso 3. Se definen los grados de libertad de todos los nudos. La figura 12.3 muestra los grados de libertad de los diferentes conjuntos de nudos. Los nudos de las esquinas son fijos, por tanto, no tienen grados de libertad. Los nudos frontera que no son esquinas tienen un grado de libertad, lo que se muestra mediante una flecha, que indica las direcciones disponibles para sus posibles movimientos. Finalmente, los nudos interiores, tienen dos grados de libertad, indicados mediante dos flechas, que muestran sus posibles movimientos.

Paso 4. Se minimiza la función objetivo Q sujeta a las restricciones frontera, y se obtienen las redes proyectadas (véase la figura 12.4).

Para evitar el solapamiento de los triángulos, se puede imponer que el área total de la red proyectada sea constante e igual a la inicial. Esta sencilla restricción evita los problemas de estabilidad provocados por el solapamiento de los triángulos.

Ejemplo 12.2 (triangulación de superficies). Considérese la superficie cuya ecuación es:

$$z = \frac{xy(x^2 - y^2)}{x^2 + y^2}$$

definida sobre el cuadrado

$$\{(x, y) | -2 \leq x \leq 2;\ -2 \leq y \leq 2\}$$

Al aplicar el método presentado con una red rectangular de tamaño $n = 12$, y

$$(x_{min}, y_{min}) = (-2, -2);\ \ (x_{max}, y_{max}) = (2, 2)$$

Tabla 12.3: Definición paramétrica de una red rectangular

	Coordenadas del nudo k
x_k	$x_{min} + floor((k-1)/n))(x_{max} - x_{min})/(n-1)$
y_k	$y_{min} + mod(k-1,n)(y_{max} - y_{min})/(n-1)$
	Nudos del elemento i para i par
Primer nudo	$(\text{floor}((i-1)/(2(n-1))))n+\text{floor}(\text{mod}((i-1),(2(n-1)))/2+1)$
Segundo node	$(\text{floor}((i-1)/(2(n-1))))n+\text{floor}(\text{mod}((i-1),(2(n-1)))/2+1)+1$
Tercer nudo	$(\text{floor}((i-1)/(2(n-1))))n+\text{floor}(\text{mod}((i-1),(2(n-1)))/2+1)+n+1$
	Nudos del elemento i para i impar
Primer nudo	$(\text{floor}((i-1)/(2(n-1))))n+\text{floor}(\text{mod}((i-1),(2(n-1)))/2+1)$
Segundo nudo	$(\text{floor}(i-1)/(2(n-1))))n+\text{floor}(\text{mod}((i-1),(2(n-1)))/2+1)+n$
Tercer nudo	$(\text{floor}((i-1)/(2(n-1))))n+\text{floor}(\text{mod}((i-1),(2(n-1)))/2+1)+n+1$
	Elementos frontera
Cara inferior	$n(j-1)+1;\ j=1,2,\ldots,n$
Cara superior	$nj;\ j=1,2,\ldots,n$
Cara derecha	$n^2-n+j;\ j=1,2,\ldots,n$
Cara izquierda	$j;\ j=1,2,\ldots,n$

se obtiene la superficie triangular de la figura 12.4. Obsérvese que los triángulos de la proyección vertical se han ajustado, para obtener triángulos 3D de área parecida.

El fichero GAMS siguiente resuelve el problema planteado:

■

```
$title Red

SETS
I numero de elementos /1*242/
J numero de nudos por elemento /1*3/
K numero de nudos /1*144/
N numero de nudos por fila /1*12/
D dimension espacial /1*2/
XFIXED(K) nudos con X fijo
YFIXED(K) nudos con Y fijo;

ALIAS(K,K1,K2);


PARAMETER
t numero de cuadrados por dimension
m numero de elementos por columna
xmin valor minimo de la coordenada  x
xmax valor maximo de la coordenada  x
ymin valor minimo de la coordenada  y
ymax valor maximo de la coordenada  y
COORD(K,D) coordenadas de los nudos de la red inicial;

t=card(N);
```

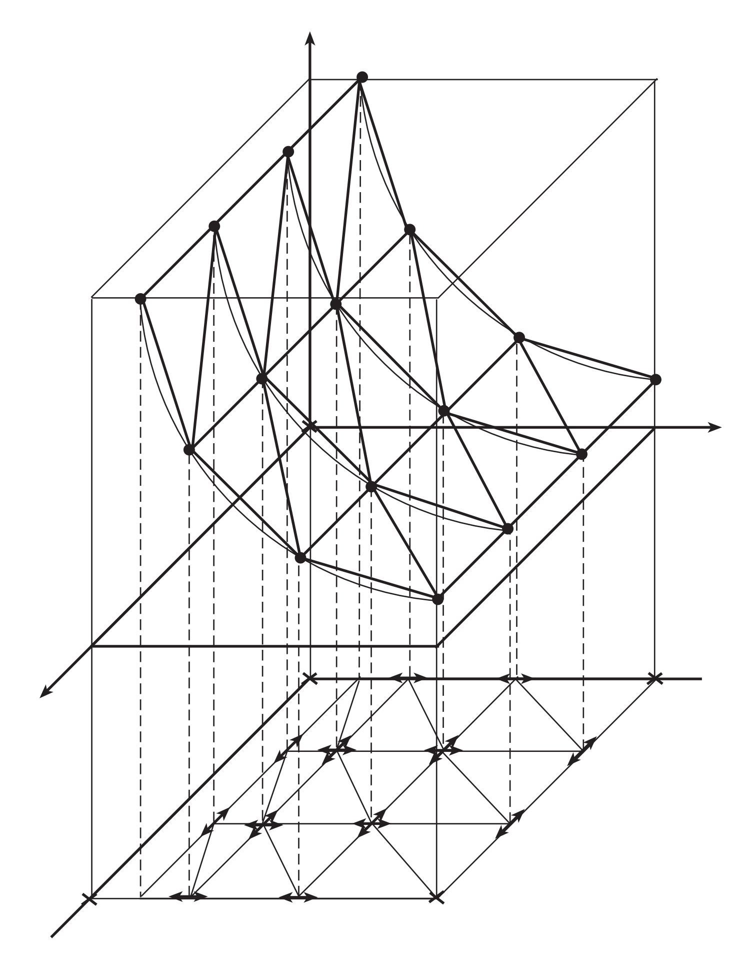


Figura 12.3: Ilustración de cómo se genera una red, mostrando los grados de libertad de los nudos.

```
xmin=-2;
xmax=2;
ymin=-2;
ymax=2;

m=2*t-2;

XFIXED(K)=no;
loop(N,
XFIXED(K)$(ord(K) eq ord(N))=yes;
XFIXED(K)$(ord(K) eq ord(N)+t*t-t)=yes;
YFIXED(K)$(ord(K) eq t*ord(N))=yes;
YFIXED(K)$(ord(K) eq (ord(N)-1)*t+1)=yes;
);

COORD(K,'1')=xmin+floor((ord(K)-1)/t)*(xmax-xmin)/(t-1);
```

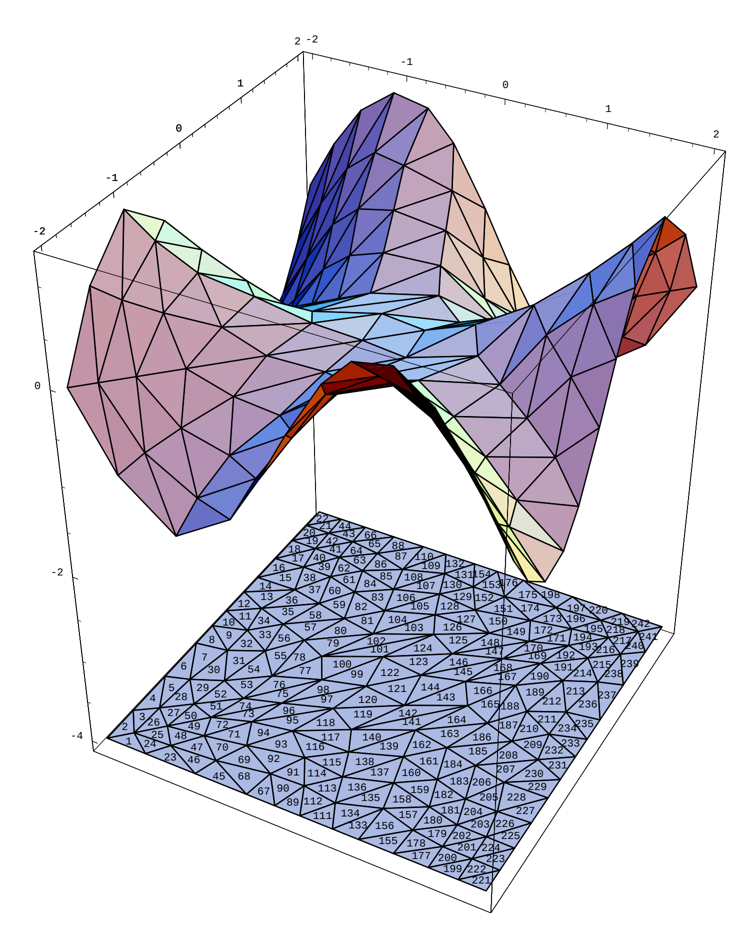


Figura 12.4: Ejemplo de una superficie triangulada mostrando la red básica y la red que obtiene los triángulos cuyas áreas tienen la menor varianza.

```
COORD(K,'2')=ymin+mod(ord(K)-1,t)*(ymax-ymin)/(t-1);

PARAMETER

EL(I,J);

loop(I,
if(mod(ord(I),2) eq 0,
EL(I,'1')=(floor((ord(I)-1)/m))*t+floor(mod((ord(I)-1),m)/2+1);

EL(I,'2')=(floor((ord(I)-1)/m))*t+floor(mod((ord(I)-1),m)/2+1)+1;

EL(I,'3')=(floor((ord(I)-1)/m))*t+floor(mod((ord(I)-1),m)/2+1)+t+1;
```

```
else
 EL(I,'1')=(floor((ord(I)-1)/m))*t+floor(mod((ord(I)-1),m)/2+1);

EL(I,'2')=(floor((ord(I)-1)/m))*t+floor(mod((ord(I)-1),m)/2+1)+t;

EL(I,'3')=(floor((ord(I)-1)/m))*t+floor(mod((ord(I)-1),m)/2+1)+t+1;
);
);

VARIABLES
z1 funcion a optimizar
X(K) coordenada del nudo K
Y(K) coordenada del nudo K
Z(K) coordenada del nudo K
area(I) area del elemento I
mean area media de los elementos;

X.l(K)=COORD(K,'1');
Y.l(K)=COORD(K,'2');

X.fx(XFIXED)=COORD(XFIXED,'1');
Y.fx(YFIXED)=COORD(YFIXED,'2');

EQUATIONS
Q funcion objetivo a optimizar
Carea(I)
Cf(K) puntos en la superficie
Cmean;

Q..z1=e=sum(I,sqr(area(I)-mean))/card(I);

Carea(I)..area(I)=e=sum((K,K1,K2)$((EL(I,'1')
   =ord(K)) y (EL(I,'2')=ord(K1)) y (EL(I,'3')=ord(K2))),
sqr((Y(K1)-Y(K))*(Z(K2)-Z(K))-(Y(K2)-Y(K))*(Z(K1)-Z(K)))+
sqr((Z(K1)-Z(K))*(X(K2)-X(K))-(Z(K2)-Z(K))*(X(K1)-X(K)))+
sqr((X(K1)-X(K))*(Y(K2)-Y(K))-(X(K2)-X(K))*(Y(K1)-Y(K))));
Cf(K)..Z(K)=e=(X(K)*Y(K)*(sqr(X(K))-sqr(Y(K))))/(sqr(X(K))+sqr(Y(K)));
Cmean..mean=e=sum(I,area(I))/card(I);

MODEL redes /ALL/;

SOLVE redes USING nlp MINIMIZING z1;

file out/redes3.out/;
put out;


if((card(I) ne (2*(t-1)*(t-1))),
put "Error, card(I) debe ser igual a ",(2*(t-1)*(t-1)):8:0/;
else
if((card(K) ne t*t),
put "Error, card(K) debe ser igual a ",(t*t):8:0/;
else
put "z1=",z1.l:12:8," media=",mean.l:12:8, " cpu=",redes.resusd:9:2,
     " modelstat=",redes.modelstat:3:0," solvestat=",redes.solvestat:3:0/;

loop(K,
put "X(",K.tl:3,")=",COORD(K,'1'):12:8,"
```

```
Y(",K.tl:3,")=",COORD(K,'2'):12:8/;
);

loop(I,
put "I=",i.tl:3," Nudos=",EL(I,'1'):5:0," ",EL(I,'2'):5:0,"
",EL(I,'3'):5:0/;
);

loop(K,
put "K=",K.tl:3," X=",X.l(K):12:6," Y=",Y.l(K):12:6," Z=",Z.l(K):12:6/;
);

loop(I,
put "I=",I.tl:3," area=",area.l(I):12:6/;
);

put "COORDENADAS"/;
put "{";

loop(K,
if(ord(K)<card(K),
put "{",X.l(K):12:8,",",Y.l(K):12:8,",",Z.l(K):12:8,"},"/;
else
put "{",X.l(K):12:8,",",Y.l(K):12:8,",",Z.l(K):12:8,"}}"/;
);
);

put "ELEMENTOS"/;
put "{";

loop(I,
if(ord(I)<card(I),
put
"{",EL(I,'1'):5:0,",",EL(I,'2'):5:0,",",EL(I,'3'):5:0,"},"/;
else
put
"{",EL(I,'1'):5:0,",",EL(I,'2'):5:0,",",EL(I,'3'):5:0,"}}"/;
););););
```

12.3 Aplicaciones a la probabilidad

En este apartado se presentan algunas aplicaciones relevantes de probabilidad. Se trata el problema de la compatibilidad de las distribuciones condicionales, que surge en muchos campos de la ingeniería, como por ejemplo la inteligencia artificial y los sistemas expertos. La evaluación de probabilidades es crucial en los modelos basados en probabilidad que se utilizan en estos campos. Esta evaluación se puede realizar considerando la distribución conjunta de todas las variables involucradas o, sencillamente, considerando los distintos conjuntos de distribuciones condicionales y/o marginales, ya que tienen menor dimensión. Sin embargo, las probabilidades condicionales están estrechamente relacionadas y, por tanto, no se pueden asignar libremente. Incluso para un experto en estos temas, existen muchas posibilidades de incumplir los axiomas de probabilidad durante la asignación. Por tanto, en la práctica, el problema de la compati-

bilidad debe abordarse con la ayuda de un computador. En este apartado, se explica cómo se puede resolver este problema mediante técnicas de programación lineal (en Arnold et al. [1, 2, 3, 4, 5] se puede encontrar un tratamiento más completo del problema).

12.3.1 Compatibilidad de matrices de probabilidad condicional

Sea una variable bidimensional discreta y aleatoria (X, Y), con dominio $N = \{(x_i, y_j)|i = 1, 2, \ldots, I; j = 1, 2, \ldots, J\}$, donde X e Y pueden tomar los valores $x_1, x_2, \ldots x_I$ y $y_1, y_2, \ldots, y_J$, respectivamente.

Sea **P** su matriz de probabilidad conjunta con $p_{ij} = \text{Prob}[X = x_i, Y = y_j]$. Los vectores columna

$$\begin{array}{rcl} p_{i.} & = & \sum_j^J p_{ij} \\ p_{.j} & = & \sum_i^I p_{ij} \end{array}$$

donde el punto se refiere a la suma en el índice correspondiente, dan las probabilidades de X e Y, respectivamente, ignorando la otra variable, y se denominan probabilidades marginales de X e Y, respectivamente.

Las matrices de probabilidad condicional **A** y **B** se definen como

$$a_{ij} = \text{Prob}[X = x_i|Y = y_j] = \frac{p_{ij}}{p_{.j}} \tag{12.15}$$

y

$$b_{ij} = \text{Prob}[Y = y_j|X = x_i] = \frac{p_{ij}}{p_{i.}} \tag{12.16}$$

respectivamente, y representan las probabilidades de una variable una vez se ha fijado la otra.

Obsérvese que **A** y **B** sólo existen si

$$\text{Prob}[Y = y_j] = \sum_i^I p_{ij} = p_{.j} \neq 0$$

y

$$\text{Prob}[X = x_i] = \sum_j^J p_{ij} = p_{i.} \neq 0$$

respectivamente.

A partir de (12.15) y (12.16) se deduce que las columnas de **A** y las filas de **B** deben sumar 1. Por tanto, las matrices de probabilidad condicional **A** y **B** deben estar formadas por elementos no negativos tales que las columnas de **A** y las filas de **B** sumen 1.

Si las probabilidades conjuntas p_{ij} son conocidas, entonces se pueden obtener las matrices de probabilidad condicional **A** y **B** mediante (12.15) y (12.16),

respectivamente. Sin embargo, la obtención de la matriz **P** a partir de **A** y **B** no es una tarea sencilla. De hecho, **A** y **B** pueden elegirse de forma que no exista un matriz **P** que satisfaga (12.15) y (12.16). En este caso se dice que **A** y **B** son incompatibles. Si, por el contrario, existe la matriz **P**, entonces **A** y **B** son compatibles.

En consecuencia, **A** y **B** son compatibles si y sólo si existe una distribución conjunta **P** con **A** y **B** como sus matrices de probabilidad condicional, es decir, de tal forma que

$$p_{ij} = a_{ij} p_{.j}; \ \forall i, j \tag{12.17}$$

$$p_{ij} = b_{ij} p_{i.}; \ \forall i, j \tag{12.18}$$

$$p_{ij} \geq 0, \quad \forall i, j, \tag{12.19}$$

$$\sum_{i=1}^{I} \sum_{j=1}^{J} p_{ij} = 1 \tag{12.20}$$

A continuación se presentan tres métodos para determinar la compatibilidad de **A** y **B**:

Método 1 (basado en la matriz de probabilidad conjunta P). Se busca la matriz de probabilidad conjunta **P** que satisfaga

$$\begin{array}{rcl} p_{ij} - a_{ij} \sum\limits_{i=1}^{I} p_{ij} & = & 0, \quad \forall i, j \\ p_{ij} - b_{ij} \sum\limits_{j=1}^{J} p_{ij} & = & 0, \quad \forall i, j \\ \sum\limits_{i=1}^{I} \sum\limits_{j=1}^{J} p_{ij} & = & 1 \\ p_{ij} & \geq & 0, \quad \forall i, j \end{array} \tag{12.21}$$

Con este método se busca **P** directamente, y esto supone $2|N| + 1$ ecuaciones con $|N|$ variables, donde $|N|$ es el cardinal del conjunto dominio N de (X, Y).

Método 2 (basado en las dos matrices de probabilidades marginales). Se buscan los vectores de probabilidad $\boldsymbol{\tau}$ y $\boldsymbol{\eta}$ que satisfagan

$$\begin{array}{rcl} \eta_j a_{ij} - \tau_i b_{ij} & = & 0, \quad \forall i, j \\ \sum\limits_{i=1}^{I} \tau_i & = & 1 \\ \sum\limits_{j=1}^{J} \eta_j & = & 1 \\ \tau_i \geq 0, \forall i, \quad \eta_j & \geq & 0, \quad \forall j \end{array} \tag{12.22}$$

Con este método se buscan los dos vectores de probabilidades marginales, $\boldsymbol{\tau}$ y $\boldsymbol{\eta}$, que, combinados con **A** y **B**, dan lugar a **P** ($p_{ij} = \eta_j a_{ij} = \tau_i b_{ij}$). Esto supone $|N| + 2$ ecuaciones con $I + J$ variables.

Método 3 (basado en una matriz de probabilidad marginal). Se busca un vector de probabilidad $\boldsymbol{\tau}$ que satisfaga

$$\begin{array}{rcl} a_{ij}\sum_{k=1}^{I}\tau_k b_{kj} - \tau_i b_{ij} & = & 0, \ \ \forall i,j \\ \sum_{i=1}^{I}\tau_i & = & 1 \\ \tau_i & \geq & 0, \ \ \forall i \end{array} \tag{12.23}$$

En este método, se busca la función de probabilidad marginal de X, $\boldsymbol{\tau}$ que, combinada con $\mathbf{B}$, da lugar a la conjunta $\mathbf{P}$ ($p_{ij} = \tau_i b_{ij}$). Este sistema tiene asociadas $|N| + 1$ ecuaciones con I variables.

Los tres métodos obtienen un sistema de ecuaciones lineales que se resuelve sujeto a restricciones de no negatividad. Sin embargo, puesto que el sistema (12.23) es el que presenta menor número de variables, se suele utilizar éste para resolver este tipo de problemas.

Para buscar soluciones factibles basta optimizar cualquier función, por ejemplo, un elemento cualquiera de las matrices de probabilidad.

Ejemplo 12.3 (compatibilidad). Sean las matrices $\mathbf{A}$ y $\mathbf{B}$:

$$\mathbf{A} = \begin{pmatrix} 0.0667 & 0.1905 & 0.3750 & 0.1176 & 0.0769 & 0.1111 \\ 0.1333 & 0.1905 & 0.0000 & 0.1765 & 0.3846 & 0.3333 \\ 0.3333 & 0.1905 & 0.1250 & 0.1765 & 0.0769 & 0.1111 \\ 0.2000 & 0.0952 & 0.2500 & 0.1176 & 0.0769 & 0.1111 \\ 0.2000 & 0.1905 & 0.1250 & 0.1765 & 0.3077 & 0.2222 \\ 0.0667 & 0.1429 & 0.1250 & 0.2353 & 0.0769 & 0.1111 \end{pmatrix} \tag{12.24}$$

$$\mathbf{B} = \begin{pmatrix} 0.0833 & 0.3333 & 0.2500 & 0.1667 & 0.0833 & 0.0833 \\ 0.1176 & 0.2353 & 0.0000 & 0.1765 & 0.2941 & 0.1765 \\ 0.3333 & 0.2667 & 0.0667 & 0.2000 & 0.0667 & 0.0667 \\ 0.2727 & 0.1818 & 0.1818 & 0.1818 & 0.0909 & 0.0909 \\ 0.1765 & 0.2353 & 0.0588 & 0.1765 & 0.2353 & 0.1176 \\ 0.0909 & 0.2727 & 0.0909 & 0.3636 & 0.0909 & 0.0909 \end{pmatrix} \tag{12.25}$$

El código GAMS que sigue genera un matriz de probabilidad conjunta aleatoria y las correspondientes condicionales. Además, obtiene la probabilidad conjunta a partir de A y B mediante los métodos 1–3.

```
$title Metodo I de Compatibilidad

SETS
I numero de filas/1*6/
J numero de columnas/1*6/;

ALIAS(I,I1);
ALIAS(J,J1);

PARAMETER
PP(I,J) Matriz de probabilidad conjunta usada para generar condicionales
```

```
A(I,J) probabilidad condicional de X dado Y
B(I,J) probabilidad condicional de Y dado X;

PP(I,J)=round(uniform(0,5));
PP(I,J)=PP(I,J)/sum((I1,J1),PP(I1,J1));
A(I,J)=PP(I,J)/sum(I1,PP(I1,J));
B(I,J)=PP(I,J)/sum(J1,PP(I,J1));

VARIABLE
z valor de la funcion objetivo;

POSITIVE VARIABLES
P(I,J) Probabilidad conjunta buscada
TAU(I) Probabilidad marginal de X
ETA(J) Probabilidad marginal de Y;

EQUATIONS
ZDEF1 Funcion a optimizar en el modelo 1
ZDEF2 Funcion a optimizar en el modelo 2
ZDEF3 Funcion a optimizar en el modelo 3
CONST11(I,J)
CONST12(I,J)
CONST2(I,J)
CONST3(I,J)
NORM1 P debe sumar uno
NORM2 TAU debe sumar uno
NORM3 ETA debe sumar uno;

ZDEF1..z=e=P('1','1');
ZDEF2..z=e=TAU('1');
CONST11(I,J)..P(I,J)-A(I,J)*sum(I1,P(I1,J))=e=0;
CONST12(I,J)..P(I,J)-B(I,J)*sum(J1,P(I,J1))=e=0;
CONST2(I,J)..ETA(J)*A(I,J)-TAU(I)*B(I,J)=e=0;
CONST3(I,J)..A(I,J)*sum(I1,TAU(I1)*B(I1,J))-TAU(I)*B(I,J)=e=0;
NORM1..sum((I,J),P(I,J))=e=1;
NORM2..sum(I,TAU(I))=e=1;
NORM3..sum(J,ETA(J))=e=1;

MODEL model1/ZDEF1,CONST11,CONST12,NORM1/;
MODEL model2/ZDEF2,CONST2,NORM2,NORM3/;
MODEL model3/ZDEF2,CONST3,NORM2/;

file out/comp1.out/;
put out;

put "Matriz Inicial PP:"/;
loop(I,loop(J,put PP(I,J):7:4;);put ""/;);
put "Matriz A:"/;
loop(I,loop(J,put A(I,J):7:4;);put ""/;);
put "Matriz B:"/;
loop(I,loop(J,put B(I,J):7:4;);put ""/;);

SOLVE model1 USING lp MINIMIZING z;
put "Matriz Final P Obtenida por el Modelo 1:"/;
loop(I,loop(J,put P.l(I,J):7:4;);put ""/;);
put "z=",z.l," modelstat=",model1.modelstat," solvestat=",model1.solvestat/;
SOLVE model2 USING lp MINIMIZING z;
```

```
put "Matriz Final P Obtenida por el Modelo 2:"/;
loop(I,loop(J,put (TAU.l(I)*B(I,J)):7:4;);put ""/;);
put "z=",z.l," modelstat=",model2.modelstat," solvestat=",model2.solvestat/;
SOLVE model3 USING lp MINIMIZING z;
put "Matriz Final P Obtenida por el Modelo 3:"/;
loop(I,loop(J,put (TAU.l(I)*B(I,J)):7:4;);put ""/;);
put "z=",z.l," modelstat=",model3.modelstat," solvestat=",model3.solvestat/;
```

La solución de este problema es la matriz de probabilidad conjunta

$$\mathbf{P} = \begin{pmatrix} 0.0120 & 0.0482 & 0.0361 & 0.0241 & 0.0120 & 0.0120 \\ 0.0241 & 0.0482 & 0.0000 & 0.0361 & 0.0602 & 0.0361 \\ 0.0602 & 0.0482 & 0.0120 & 0.0361 & 0.0120 & 0.0120 \\ 0.0361 & 0.0241 & 0.0241 & 0.0241 & 0.0120 & 0.0120 \\ 0.0361 & 0.0482 & 0.0120 & 0.0361 & 0.0482 & 0.0241 \\ 0.0120 & 0.0361 & 0.0120 & 0.0482 & 0.0120 & 0.0120 \end{pmatrix} \tag{12.26}$$

Dado que el problema de programación lineal tiene solución, se tiene que las matrices de probabilidad condicional **A** y **B** son compatibles. ■

Ejemplo 12.4 (incompatibilidad). Sean las matrices **A** y **B**:

$$\mathbf{A} = \begin{pmatrix} \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & 0 & \frac{1}{2} \end{pmatrix}, \tag{12.27}$$

$$\mathbf{B} = \begin{pmatrix} \frac{1}{3} & \frac{2}{3} & 0 \\ 0 & \frac{1}{3} & \frac{2}{3} \\ \frac{1}{3} & 0 & \frac{2}{3} \end{pmatrix}. \tag{12.28}$$

Dado que el correspondiente problema de programación lineal no tiene solución, se tiene que las matrices de probabilidad condicional **A** y **B** son incompatibles

Se recomienda al lector que compruebe la infactibilidad del programa modificando el código GAMS presentado previamente. ■

12.3.2 Cuasi-compatibilidad

En el apartado 12.3.1 se describieron 3 formulaciones de las ecuaciones lineales para obtener una matriz **P** compatible con la matrices de probabilidad condicional **A** y **B**. Si, en lugar de compatibilidad exacta, se requiere una compatibilidad aproximada, se pueden reemplazar los métodos 1–3 por las siguientes versiones revisadas:

Método 1 revisado. Busca una matriz de probabilidad **P** para minimizar ϵ

sujeto a

$$\begin{array}{rcl} |p_{ij} - a_{ij} \sum\limits_{i=1}^{I} p_{ij}| & \leq & \epsilon\, \gamma_{ij}, \ \ \forall i, j \\ |p_{ij} - b_{ij} \sum\limits_{j=1}^{J} p_{ij}| & \leq & \epsilon\, \gamma_{ij}, \ \ \forall i, j \\ \sum\limits_{i=1}^{I} \sum\limits_{j=1}^{J} p_{ij} & = & 1 \\ p_{ij} & \geq & 0, \ \ \forall i, j \end{array} \qquad (12.29)$$

Método 2 revisado. Busca dos vectores de probabilidad, $\boldsymbol{\tau}$ y $\boldsymbol{\eta}$, para minimizar ϵ sujeto a

$$\begin{array}{rcl} |\eta_j a_{ij} - \tau_i b_{ij}| & \leq & \epsilon\, \gamma_{ij}, \ \ \forall i, j \\ \sum\limits_{i=1}^{I} \tau_i & = & 1 \\ \sum\limits_{j=1}^{J} \eta_j & = & 1 \\ \tau_i \geq 0, \eta_j & \geq & 0, \forall i, j \end{array} \qquad (12.30)$$

Método 3 revisado. Busca un vector de probabilidad $\boldsymbol{\tau}$ para minimizar ϵ sujeto a

$$\begin{array}{rcl} |a_{ij} \sum\limits_{i=1}^{I} \tau_i b_{ij} - \tau_i b_{ij}| & \leq & \epsilon\, \gamma_{ij}, \ \ \forall i, j \\ \sum\limits_{i=1}^{I} \tau_i & = & 1 \\ \tau_i & \geq & 0, \ \ \forall i \end{array} \qquad (12.31)$$

donde γ_{ij} son parámetros que miden la importancia relativa de los errores que se está dispuesto a aceptar para los valores p_{ij} de probabilidades conjuntas, y ϵ mide el grado global de incompatibilidad. Obsérvese que si $\epsilon = 0$, se tiene compatibilidad; si no, no se tiene.

El principal objetivo consiste en obtener las matrices **P**, y los vectores $\boldsymbol{\tau}$ y $\boldsymbol{\eta}$ que minimicen ϵ, es decir, el grado de incompatibilidad.

El problema asociado es un PPL, puesto que cada restricción que contiene la función valor absoluto en (12.29)–(12.31) se puede reemplazar por dos restricciones lineales de desigualdad (véase el capítulo 13). Por tanto, todas las restricciones se pueden escribir como restricciones lineales.

Ejemplo 12.5 (ejemplo compatible). Sean las matrices **A** y **B**:

$$A = \begin{pmatrix} 0.1429 & 0.3333 & 0.2727 & 0.1667 \\ 0.1429 & 0.0833 & 0.1818 & 0.3333 \\ 0.0000 & 0.2500 & 0.4545 & 0.2500 \\ 0.7143 & 0.3333 & 0.0909 & 0.2500 \end{pmatrix} \qquad (12.32)$$

$$B = \begin{pmatrix} 0.1000 & 0.4000 & 0.3000 & 0.2000 \\ 0.1250 & 0.1250 & 0.2500 & 0.5000 \\ 0.0000 & 0.2727 & 0.4545 & 0.2727 \\ 0.3846 & 0.3077 & 0.0769 & 0.2308 \end{pmatrix} \qquad (12.33)$$

El código GAMS que permite obtener la matriz de probabilidad conjunta, y resuelve el problema mediante los métodos revisados 1–3 es:

```
$title Metodos 1, 2 y 3 de Compatibilidad

SETS
I numero de filas/1*4/
J numero de columnas/1*4/;

ALIAS(I,I1);
ALIAS(J,J1);

PARAMETER
PP(I,J) Matriz de probabilidad conjunta usada para obtener las condicionales
A(I,J) probabilidad condicional de X dado Y
B(I,J) probabilidad condicional de Y dado X;

PP(I,J)=round(uniform(0,5));
PP(I,J)=PP(I,J)/sum((I1,J1),PP(I1,J1));
A(I,J)=PP(I,J)/sum(I1,PP(I1,J));
B(I,J)=PP(I,J)/sum(J1,PP(I,J1));

* La linea que sigue modifica la matriz B para obtener incompatibilidad
* B('1',J)=0.25;

VARIABLE
z valor de la funcion objetivo
epsilon Error maximo en la probabilidad conjunta;

POSITIVE VARIABLES
P(I,J) Probabilidad conjunta de X e Y
TAU(I) Probabilidad marginal de X
ETA(J) Probabilidad marginal de Y;

EQUATIONS
ZDEF1 Funcion a optimizar en el modelo 1
ZDEF2 Funcion a optimizar en el modelo 2
ZDEF3 Funcion a optimizar en el modelo 3
CONST111(I,J)
CONST112(I,J)
CONST121(I,J)
CONST122(I,J)
CONST21(I,J)
CONST22(I,J)
CONST31(I,J)
CONST32(I,J)
NORM1 P suma uno
NORM2 TAU suma uno
NORM3 ETA suma uno;

ZDEF1..z=e=epsilon;
CONST111(I,J)..P(I,J)-A(I,J)*sum(I1,P(I1,J))=l=epsilon;
CONST112(I,J)..-epsilon=l=P(I,J)-A(I,J)*sum(I1,P(I1,J));
CONST121(I,J)..P(I,J)-B(I,J)*sum(J1,P(I,J1))=l=epsilon;
CONST122(I,J)..-epsilon=l=P(I,J)-B(I,J)*sum(J1,P(I,J1));
CONST21(I,J)..ETA(J)*A(I,J)-TAU(I)*B(I,J)=l=epsilon;
CONST22(I,J)..-epsilon=l=ETA(J)*A(I,J)-TAU(I)*B(I,J);
CONST31(I,J)..A(I,J)*sum(I1,TAU(I1)*B(I1,J))-TAU(I)*B(I,J)=l=epsilon;
```

```
CONST32(I,J)..-epsilon=l=A(I,J)*sum(I1,TAU(I1)*B(I1,J))-TAU(I)*B(I,J);
NORM1..sum((I,J),P(I,J))=e=1;
NORM2..sum(I,TAU(I))=e=1;
NORM3..sum(J,ETA(J))=e=1;

MODEL model1/ZDEF1,CONST111,CONST112,CONST121,CONST122,NORM1/;
MODEL model2/ZDEF1,CONST21,CONST22,NORM2,NORM3/;
MODEL model3/ZDEF1,CONST31,CONST32,NORM2/;

file out/comp1.out/;
put out;

put "Matriz inicial PP:"/;
loop(I,loop(J,put PP(I,J):7:4," & ";);put ""/;);
put "Matriz A:"/;
loop(I,loop(J,put A(I,J):7:4," & ";);put ""/;);
put "Matriz B:"/;
loop(I,loop(J,put B(I,J):7:4," & ";);put ""/;);

SOLVE model1 USING lp MINIMIZING z;
put "Matriz final P obtenida por el modelo 1:"/;
loop(I,loop(J,put P.l(I,J):7:4," & ";);put ""/;);
put "z=",z.l:12:8," modelstat=",model1.modelstat,"
solvestat=",model1.solvestat/;
SOLVE model2 USING lp MINIMIZING z;
put "Matriz final P obtenida por el modelo 2:"/;
loop(I,loop(J,put (TAU.l(I)*B(I,J)):7:4," & ";);put ""/;);
put "z=",z.l:12:8," modelstat=",model2.modelstat,"
solvestat=",model2.solvestat/;
SOLVE model3 USING lp MINIMIZING z;
put "Matriz final P obtenida por el modelo 3:"/;
loop(I,loop(J,put (TAU.l(I)*B(I,J)):7:4," & ";);put ""/;);
put "z=",z.l:12:8," modelstat=",model3.modelstat,"
solvestat=",model3.solvestat/;
```

La solución de este problema, que coincide con la que se obtiene mediante los tres métodos, es $\epsilon = 0$ y la matriz de probabilidad conjunta es

$$\mathbf{P} = \begin{pmatrix} 0.0238 & 0.0952 & 0.0714 & 0.0476 \\ 0.0238 & 0.0238 & 0.0476 & 0.0952 \\ 0.0000 & 0.0714 & 0.1190 & 0.0714 \\ 0.1190 & 0.0952 & 0.0238 & 0.0714 \end{pmatrix} \tag{12.34}$$

■

Ejemplo 12.6 (ejemplo incompatible). Si se modifica la primera fila de la matriz B en el ejemplo 12.5 [en (12.33)], dejando la línea de comentario sin asterisco, se obtiene el siguiente fichero de salida:

```
Matriz inicial PP:
 0.0238 &  0.0952 &  0.0714 &  0.0476 &
 0.0238 &  0.0238 &  0.0476 &  0.0952 &
 0.0000 &  0.0714 &  0.1190 &  0.0714 &
 0.1190 &  0.0952 &  0.0238 &  0.0714 &
```

```
Matriz A:
 0.1429 &  0.3333 &  0.2727 &  0.1667 &
 0.1429 &  0.0833 &  0.1818 &  0.3333 &
 0.0000 &  0.2500 &  0.4545 &  0.2500 &
 0.7143 &  0.3333 &  0.0909 &  0.2500 &
Matriz B:
 0.2500 &  0.2500 &  0.2500 &  0.2500 &
 0.1250 &  0.1250 &  0.2500 &  0.5000 &
 0.0000 &  0.2727 &  0.4545 &  0.2727 &
 0.3846 &  0.3077 &  0.0769 &  0.2308 &
Matriz final P obtenida por el modelo 1:
 0.0403 &  0.0630 &  0.0539 &  0.0494 &
 0.0289 &  0.0165 &  0.0549 &  0.1231 &
 0.0000 &  0.0579 &  0.1177 &  0.0784 &
 0.1329 &  0.0858 &  0.0129 &  0.0843 &
z=  0.01139203 modelstat=       1.00 solvestat=       1.00
Matriz final P obtenida por el modelo 2:
 0.0478 &  0.0478 &  0.0478 &  0.0478 &
 0.0335 &  0.0335 &  0.0670 &  0.1340 &
 0.0000 &  0.0678 &  0.1131 &  0.0678 &
 0.1124 &  0.0899 &  0.0225 &  0.0674 &
z=  0.02107728 modelstat=       1.00 solvestat=       1.00
Matriz final P obtenida por el modelo 3:
 0.0559 &  0.0559 &  0.0559 &  0.0559 &
 0.0309 &  0.0309 &  0.0618 &  0.1237 &
 0.0000 &  0.0524 &  0.0873 &  0.0524 &
 0.1295 &  0.1036 &  0.0259 &  0.0777 &
z=  0.02502453 modelstat=       1.00 solvestat=       1.00
```

que muestra las matrices iniciales P, A, y B y las matrices P obtenidas mediante los métodos 1 a 3. Obsérvese que las matrices A y B son incompatibles, y que los valores de ϵ para los métodos 1, 2, y 3 son 0.01139203, 0.02107728 y 0.02502453, respectivamente. ∎

12.4 Modelos de regresión

Se aborda a continuación el problema de la regresión. Los métodos de mínimos cuadrados están en auge debido al buen comportamiento matemático de las funciones cuadráticas asociadas a los mismos, que son derivables y dan lugar a sistemas de ecuaciones sencillos. Sin embargo, existen otros métodos útiles, como los basados en valores absolutos o máximos, que a pesar de su relevancia, no suelen emplearse debido a que las funciones asociadas no son derivables. Afortunadamente, las técnicas de programación lineal permiten abordar estos problemas de manera eficiente, tal y como se describe en este apartado.

Considérese un modelo lineal

$$\mathbf{y} = \mathbf{Z}\boldsymbol{\beta} + \boldsymbol{\varepsilon} \tag{12.35}$$

o, el equivalente

$$y_i = \mathbf{z}_i^T\boldsymbol{\beta} + \varepsilon_i, \; i = 1, \ldots, n \tag{12.36}$$

donde $\mathbf{y} = (y_1, \ldots, y_n)^T$ es un vector de dimensión $n \times 1$, de variables respuesta, $\mathbf{Z}$ es una matriz de dimensión $n \times p$, de variables de predicción, $\mathbf{z}_i^T$ es la i-ésima

fila de $\mathbf{Z}$, $\boldsymbol{\beta}$ es un vector de dimensión $p \times 1$, de coeficientes o parámetros de regresión, y $\boldsymbol{\varepsilon} = (\varepsilon_1, \ldots, \varepsilon_n)^T$ es un vector de dimensión $n \times 1$, de errores aleatorios.

Los métodos más empleados para la estimación de parámetros de regresión $\boldsymbol{\beta}$ son

1. El método de los mínimos cuadrados (MC). Este método minimiza

$$\sum_{i=1}^{n} (y_i - \mathbf{z}_i^t \boldsymbol{\beta})^2 \tag{12.37}$$

Éste es el modelo de regresión estándar, y se puede resolver fácilmente mediante técnicas bien conocidas. Este método penaliza los errores grandes respecto a los errores pequeños, puesto que los errores se elevan al cuadrado; es decir, los errores grandes se amplían y los errores pequeños se reducen. Por tanto, este método debe emplearse cuando sólo tienen importancia los errores grandes y no los pequeños.

2. El método del mínimo valor absoluto (MVA). Este método minimiza (véase, por ejemplo, Arthanari y Dodge [6])

$$\sum_{i=1}^{n} |y_i - \mathbf{z}_i^t \boldsymbol{\beta}| \tag{12.38}$$

Este método minimiza la suma de las distancias entre los valores observados y las predicciones en lugar de sus cuadrados. Sin embargo, debido a la presencia de la función valor absoluto, es complicado resolver este problema mediante técnicas de regresión estándar. Este método trata todos los errores de igual manera. Por tanto, debe usarse cuando todos los errores tienen idéntica importancia. De hecho, lo que importa es la suma de los errores absolutos, no un único error.

3. El método minimax (MM). Se minimiza

$$\max_i |y_i - \mathbf{z}_i^t \boldsymbol{\beta}|, \tag{12.39}$$

Este método minimiza el máximo de las distancias entre los valores observados y las predicciones. Esta alternativa es válida si lo que importa es el error máximo. Sin embargo, debido a la presencia de la función máximo, tiene asociadas las mismas complicaciones que el método anterior.

Las estimaciones de $\boldsymbol{\beta}$ para los dos últimos métodos se pueden obtener resolviendo sencillos problemas de programación lineal. La estimación MVA de $\boldsymbol{\beta}$, $\hat{\boldsymbol{\beta}}$, se puede obtener mediante la resolución del siguiente problema de programación lineal (PPL) (véase Castillo et al. [22]). Minimizar

$$\sum_{i=1}^{n} \varepsilon_i \tag{12.40}$$

sujeto a

$$\begin{aligned} y_i - \mathbf{z}_i^t\boldsymbol{\beta} \le \varepsilon_i, &\quad i = 1, \dots, n \\ \mathbf{z}_i^t\boldsymbol{\beta} - y_i \le \varepsilon_i, &\quad i = 1, \dots, n \\ \varepsilon_i \ge 0, &\quad i = 1, \dots, n \end{aligned} \tag{12.41}$$

Análogamente, la estimación MM de $\boldsymbol{\beta}$, $\tilde{\boldsymbol{\beta}}$, se puede obtener resolviendo el siguiente PPL. Minimizar

$$\varepsilon \tag{12.42}$$

sujeto a

$$\begin{aligned} y_i - \mathbf{z}_i^t\boldsymbol{\beta} \le \varepsilon, &\quad i = 1, \dots, n \\ \mathbf{z}_i^t\boldsymbol{\beta} - y_i \le \varepsilon, &\quad i = 1, \dots, n \\ \varepsilon \ge 0 & \end{aligned} \tag{12.43}$$

Sean $\hat{\boldsymbol{\beta}}_{(i)}$ y $\tilde{\boldsymbol{\beta}}_{(i)}$, respectivamente, los estimadores MVA y MM de $\boldsymbol{\beta}$ cuando se omite la i-ésima observación, respectivamente. La influencia de la i-ésima observación sobre los estimadores MVA puede medirse mediante

$$d_i(\text{MVA}) = ||\hat{\mathbf{y}} - \hat{\mathbf{y}}_{(i)}|| \tag{12.44}$$

donde $\hat{\mathbf{y}} = \mathbf{Z}\hat{\boldsymbol{\beta}}$ es el vector de valores estimados, e $\hat{\mathbf{y}}_{(i)} = \mathbf{Z}\hat{\boldsymbol{\beta}}_{(i)}$ es el vector de valores estimados cuando se omite la i-ésima observación. Por tanto, $d_i(\text{LAV})$ es la norma de la diferencia entre los vectores de predicciones obtenidos a partir de información total y parcial, respectivamente.

Análogamente, la influencia de la i-ésima observación sobre los estimadores MM puede medirse mediante

$$d_i(\text{MM}) = ||\tilde{\mathbf{y}} - \tilde{\mathbf{y}}_{(i)}|| \tag{12.45}$$

donde $\tilde{\mathbf{y}} = \mathbf{Z}\tilde{\boldsymbol{\beta}}$ y $\tilde{\mathbf{y}}_{(i)} = \mathbf{Z}\tilde{\boldsymbol{\beta}}_{(i)}$ Finalmente, $d_i(\text{MC})$ se puede definir de manera similar.

Para calcular $d_i(\text{MVA})$ o $d_i(\text{MM})$, se necesita resolver $n+1$ problemas de programación lineal, a menos que se empleen herramientas de sensibilidad para reducir el tiempo de cálculo.

El siguiente ejemplo ilustra las distintas sensibilidades de los tres métodos respecto a *outliers*, valores muy distintos de los habituales.

Ejemplo 12.7 (robustez respecto a los outliers). Considérense los datos de la tabla 12.4, que han sido obtenidos mediante el modelo

$$Y_i = 0.2 + 0.1Z_1 - 0.3Z_2 + 0.4Z_3 - 0.2Z_4 + \epsilon_i$$

donde $\epsilon_i \sim U(0, 0.05)$ y los puntos $7, 15$ y 16 se han modificado para convertirlos en outliers mediante

$$\begin{aligned} Y_7 &= 0.2 + 0.1Z_1 - 0.3Z_2 + 0.4Z_3 - 0.2Z_4 - 0.05 \times 2 \\ Y_{15} &= 0.2 + 0.1Z_1 - 0.3Z_2 + 0.4Z_3 - 0.2Z_4 + 0.05 \times 2 \\ Y_{16} &= 0.2 + 0.1Z_1 - 0.3Z_2 + 0.4Z_3 - 0.2Z_4 - 0.05 \times 2 \end{aligned}$$

Se han utilizado los tres modelos – MC, MVA y MM, formulados en (12.37), (12.38), y (12.39), respectivamente – para ajustar los datos. El código GAMS que resuelve este ejemplo de regresión es:

```
$ title Regresion

SETS
I numero de puntos /1*30/
P numero de parametros/1*5/
J numero de modelos de regresion/1*3/
SS(I) subconjunto de datos usados en el analisis;

PARAMETER
C(P) coeficientes de regresion usados en la simulacion
/1 0.2
2 0.1
3 -0.3
4 0.4
5 -0.2/;

PARAMETER
Z(I,P) variables observadas y predicciones;

Z(I,P)=uniform(0,1);
Z(I,'1')=C('1')+sum(P$(ord(P)>1),C(P)*Z(I,P))+uniform(0,1)*0.05;
Z('7','1')=C('1')+sum(P$(ord(P)>1),C(P)*Z('7',P))-0.05*2;
Z('15','1')=C('1')+sum(P$(ord(P)>1),C(P)*Z('15',P))+0.05*2;
Z('16','1')=C('1')+sum(P$(ord(P)>1),C(P)*Z('16',P))-0.05*2;

PARAMETER
Y(I)  Valores observados
Y1(I) Valores ajustados con todos los datos
Y2(I) Valores ajustados tras eliminar un dato
z0 Valor auxiliar;

Y(I)=Z(I,'1');

POSITIVE VARIABLES
EPSILON1(I) error asociado a los datos I
EPSILON error;

VARIABLES
z1 valor de la funcion objetivo
z2 valor de la funcion objetivo
z3 valor de la funcion objetivo
BETA(P);

EQUATIONS
dz1 definicion del valor de la funcion objetivo 1
dz2 definicion del valor de la funcion objetivo 2
dz3 definicion del valor de la funcion objetivo 3
lower1(I) limite inferior del error
upper1(I) limite superior del error
lower2(I) limite inferior del error
upper2(I) limite superior del error;

dz1..z1=e=sum(SS,EPSILON1(SS));
```

```
dz2..z2=e=EPSILON;
dz3..z3=e=sum(SS,sqr(Y(SS)-sum(P,Z(SS,P)*BETA(P))));
lower1(SS)..Y(SS)-sum(P,Z(SS,P)*BETA(P))=l=EPSILON1(SS);
upper1(SS)..-Y(SS)+sum(P,Z(SS,P)*BETA(P))=l=EPSILON1(SS);
lower2(SS)..Y(SS)-sum(P,Z(SS,P)*BETA(P))=l=EPSILON;
upper2(SS)..-Y(SS)+sum(P,Z(SS,P)*BETA(P))=l=EPSILON;

MODEL regresMVA/dz1,lower1,upper1/;
MODEL regresMM/dz2,lower2,upper2/;
MODEL regresMC/dz3/;

file out/regress1.out/;
put out;

put "Datos"/;

loop(I,
put I.tl:3;
loop(P,put " & ",Z(I,P):8:4;);
put "\\"/;
);
Z(I,'1')=1.0;

loop(J,
  SS(I)=yes;
  if(ord(J) eq 1,SOLVE regresMVA USING lp MINIMIZING z1;Z0=z1.l);
  if(ord(J) eq 2,SOLVE regresMM USING lp MINIMIZING z2;Z0=z2.l);
  if(ord(J) eq 3,SOLVE regresMC USING nlp MINIMIZING z3;Z0=z3.l);
  put "Todos los puntos z0=",z0:12:8 ;
    loop(P,put " & ",BETA.l(P):10:5;);
    put "\\"/;
    Y1(I)=sum(P,Z(I,P)*BETA.l(P));
  loop(I,
    SS(I)=no;
    if(ord(J) eq 1,SOLVE regresMVA USING lp MINIMIZING z1;Z0=z1.l);
    if(ord(J) eq 2,SOLVE regresMM USING lp MINIMIZING z2;Z0=z2.l);
    if(ord(J) eq 3,SOLVE regresMC USING nlp MINIMIZING z3;Z0=z3.l);
    put "Punto eliminado ",I.tl:3," z0=",z0:12:8;
    loop(P,put " & ",BETA.l(P):10:5;);
    put "\\"/;
    Y2(I)=sum(P,Z(I,P)*BETA.l(P))-Y1(I);
    SS(I)=yes;
  );
  loop(I,put I.tl:3,Y2(I):10:6/;);
);
```

Los valores resultantes de $d_i(\text{MVA}), d_i(\text{MM})$, y $d_i(\text{MC})$ se muestran en la tabla 12.5, donde los outliers y los tres valores mayores de cada columna se muestran en negrita. Obsérvese que el método MC no permite identificar los tres outliers, mientras que el método MM no permite identificar el punto 16, y el método MVA no permite de identificarlos. Esto demuestra que el método MVA es más robusto respecto a los outliers que el método MM y, por tanto, también es más robusto que el método MC. Es interesante notar que se produce un cambio en cero de $d_i(\text{MM})$ cuando se elimina un único punto que no está en

Tabla 12.4: Puntos (datos) simulados

i	Y_i	Z_{1i}	Z_{2i}	Z_{3i}	Z_{4i}
1	0.2180	0.8433	0.5504	0.3011	0.2922
2	-0.0908	0.3498	0.8563	0.0671	0.5002
3	0.2468	0.5787	0.9911	0.7623	0.1307
4	0.3431	0.1595	0.2501	0.6689	0.4354
5	0.1473	0.3514	0.1315	0.1501	0.5891
6	0.3076	0.2308	0.6657	0.7759	0.3037
7	**0.3981**	**0.5024**	**0.1602**	**0.8725**	**0.2651**
8	0.2089	0.5940	0.7227	0.6282	0.4638
9	0.0879	0.1177	0.3142	0.0466	0.3386
10	0.3795	0.6457	0.5607	0.7700	0.2978
11	0.2259	0.7558	0.6274	0.2839	0.0864
12	-0.0147	0.6413	0.5453	0.0315	0.7924
13	0.3731	0.1757	0.5256	0.7502	0.1781
14	0.1575	0.5851	0.6212	0.3894	0.3587
15	**0.5829**	**0.2464**	**0.1305**	**0.9334**	**0.3799**
16	**0.3781**	**0.3000**	**0.1255**	**0.7489**	**0.0692**
17	0.3163	0.0051	0.2696	0.4999	0.1513
18	0.0804	0.3306	0.3169	0.3221	0.9640
19	0.3913	0.3699	0.3729	0.7720	0.3967
20	-0.0961	0.1196	0.7355	0.0554	0.5763
21	0.1868	0.0060	0.4012	0.5199	0.6289
22	0.0703	0.3961	0.2760	0.1524	0.9363
23	0.2184	0.1347	0.3861	0.3746	0.2685
24	0.1059	0.1889	0.2975	0.0746	0.4013
25	0.1961	0.3839	0.3241	0.1921	0.1124
26	0.3889	0.5114	0.0451	0.7831	0.9457
27	0.2762	0.6073	0.3625	0.5941	0.6799
28	0.2523	0.1593	0.6569	0.5239	0.1244
29	0.1535	0.2281	0.6757	0.7768	0.9325
30	0.1480	0.2971	0.1972	0.2463	0.6465

los extremos. Esto se debe a que la línea de regresión sólo está definida para dos puntos extremos (outliers o no).

La tabla 12.6 muestra las estimaciones de parámetros para los tres modelos, MVA, MM y MC, estimados usando todos los datos, y estimados eliminando una de los primeros cinco puntos o outliers. Se puede observar la sensibilidad de cada parámetro respecto a cada outlier. De nuevo, se comprueba que el método MVA es más robusto respecto a outliers (presenta pequeños cambios en la estimación del parámetro cuando se elimina un outlier) que el método MM.

■

Tabla 12.5: Valores de d_i(MVA), d_i(MM), y d_i(MC), que permiten analizar la influencia de los outliers en las predicciones

i	d_i(MVA)	d_i(MM)	d_i(MC)
1	**-0.017913**	0.000000	-0.008427
2	0.007130	0.000000	0.007199
3	0.003571	0.000000	0.006221
4	0.008645	0.000000	-0.000730
5	0.001230	0.000000	0.000278
6	-0.005584	0.000000	-0.001787
7	0.000400	**0.032893**	**0.021929**
8	0.000000	0.000000	0.001975
9	-0.001667	0.000000	-0.000668
10	-0.009012	0.000000	-0.003484
11	**-0.014035**	-0.017450	-0.010407
12	-0.011423	0.000000	-0.001843
13	-0.005477	0.000000	-0.004351
14	0.000000	0.000000	0.001429
15	-0.007005	**-0.056369**	**-0.019154**
16	0.002400	0.011714	**0.023611**
17	0.009702	0.000000	-0.003079
18	0.001470	0.017649	0.003613
19	-0.004419	0.000000	-0.001829
20	0.004550	0.000000	0.008311
21	-0.000332	0.000000	0.000371
22	-0.005133	0.000000	-0.003675
23	0.001806	0.000000	-0.000840
24	-0.002236	0.000000	-0.001722
25	0.000000	0.000000	0.001087
26	-0.005122	0.000000	-0.006582
27	0.001186	0.000000	-0.001038
28	-0.006072	0.000000	-0.004491
29	**0.014205**	**0.054685**	0.013893
30	0.000673	0.000000	0.000769

12.5 Aplicaciones a la discretización de problemas de optimización continuos

Aquellos problemas de optimización de dimensión infinita, donde funciones reemplazan a vectores en las funciones de coste, se pueden convertir en problemas de dimensión finita si se discretizan mediante cálculos aproximados de las funciones óptimas. Aunque en este libro no se aborda este tipo de problemas, es interesante mostrar cómo se pasa de problemas de dimensión infinita a

Tabla 12.6: Estimación de parámetros mediante los tres métodos, usando todos los puntos y eliminando uno de los primeros 6 puntos o outliers

i	Y_i	β_1	β_2	β_3	β_4
Estimación de parámetros para el método MVA					
Todos	0.23134	0.11741	-0.31868	0.41337	-0.22561
1	0.22128	0.09177	-0.30947	0.43343	-0.21652
2	0.22807	0.11690	-0.30427	0.40866	-0.22850
3	0.22944	0.11313	-0.31254	0.41648	-0.22957
4	0.23360	0.09454	-0.31811	0.42908	-0.22703
5	0.23074	0.12023	-0.31984	0.41182	-0.22353
6	0.23098	0.10934	-0.32361	0.40977	-0.21668
7	0.23108	0.11721	-0.31960	0.41392	-0.22401
15	0.22807	0.11690	-0.30427	0.40866	-0.22850
16	0.23291	0.11002	-0.31565	0.41743	-0.23107
Estimación de parámetros para el método MM					
Todos	0.21723	0.00519	-0.28273	0.37877	-0.09172
1	0.21723	0.00519	-0.28273	0.37877	-0.09172
2	- 0.21723	0.00519	-0.28273	0.37877	-0.09172
3	0.21723	0.00519	-0.28273	0.37877	-0.09172
4	0.21723	0.00519	-0.28273	0.37877	-0.09172
5	0.21723	0.00519	-0.28273	0.37877	-0.09172
6	0.21723	0.00519	-0.28273	0.37877	-0.09172
7	0.09010	0.13240	-0.27174	0.48689	-0.09162
15	0.14378	0.06891	-0.18892	0.38863	-0.14452
16	0.22157	-0.03975	-0.23504	0.40342	-0.14350
Estimación de parámetros para el método MC					
Todos	0.20098	0.08080	-0.26760	0.39807	-0.17825
1	0.20066	0.06789	-0.26633	0.40153	-0.17470
2	0.20061	0.07838	-0.25826	0.39316	-0.17677
3	0.19746	0.08240	-0.26064	0.40084	-0.17975
4	0.20037	0.08183	-0.26696	0.39742	-0.17827
5	0.20131	0.08090	-0.26801	0.39779	-0.17824
6	0.20173	0.08264	-0.26996	0.39625	-0.17818
7	0.20658	0.09872	-0.29242	0.41429	-0.18902
15	0.19585	0.08594	-0.24951	0.37834	-0.17624
16	0.22322	0.08249	-0.29511	0.40603	-0.20198

dimensión finita en algunos casos concretos. Estos ejemplos son muy sencillos y sólo son de interés desde la perspectiva académica, no permitiendo mostrar la complejidad asociada a casos más realistas. Con estos ejemplos se puede analizar cómo se realizan las discretizaciones y, además, se amplía el conjunto de problemas de optimización que pueden resolverse mediante las técnicas y herra-

mientas que se incluyen en este texto. Concretamente, se enfatiza el uso de una herramienta como GAMS para aproximar y encontrar soluciones óptimas de las versiones discretizadas de los problemas de optimización de dimensión infinita.

Se distinguen dos categorías principales dentro de los problemas de optimización de dimensión infinita: problemas de cálculo variacional y problemas de control óptimo. En los siguientes apartados se describen y resuelven algunos problemas incluidos en cada una de estas categorías, como son la braquistocrona y la cuerda colgante para la primera categoría y, el control óptimo para alcanzar un blanco, y de un oscilador armónico, para la segunda. En todos ellos se utiliza un esquema de discretización muy simple. Si se necesitan aproximaciones más eficientes y precisas se deben utilizar esquemas específicos de mayor complejidad. En Polak [86] y Troutman [99] se describen algunos ejemplos adicionales.

12.5.1 Problemas de cálculo variacional

Un problema variacional escalar se expresa típicamente de la siguiente manera. Minimizar

$$T(u) = \int_a^b F(t, u(t), u'(t))dt$$

sujeto a

$$u(a) = A, u(b) = B$$

donde la función u es desconocida y se supone continua.

Las restricciones con integrales del tipo

$$k = \int_a^b G(t, u(t), u'(t))dt$$

son muy comunes.

¿Cómo es la versión discretizada del problema original?. En adelante, se divide el intervalo $[a, b]$ en $n + 1$ subintervalos iguales, y se supone que las funciones factibles para el nuevo problema de optimización discretizado son afines a trozos, es decir, afines en cada subintervalo

$$[a + j\Delta, a + (j + 1)\Delta]$$

donde

$$\Delta = \frac{(b - a)}{(n + 1)}$$

Obsérvese que este tipo de funciones quedan exclusivamente determinadas mediante sus valores en los nudos

$$a + j\Delta, \quad j = 1, \dots, n$$

y, por tanto, los vectores factibles para los nuevos problemas de optimización se corresponden con estos valores. Se muestra a continuación cómo este proceso permite transformar el problema original, de dimensión infinita, en un problema de dimensión finita. La idea es que con un número de subintervalos cada

vez mayor, las soluciones de este problema de optimización discretizado se asemejan y aproximan bien, bajo ciertas condiciones, a las soluciones óptimas del problema de optimización original. Sean

$$x = (x_j), \;\; 1 \leq j \leq n. \;\; x_0 = A, \;\; x_{n+1} = B$$

los valores nodales de las funciones factibles. De esta forma la función u que se emplea para optimizar $T(u)$ es

$$u(t) = x_j + \frac{(x_{j+1} - x_j)}{\Delta} (t - a - j\Delta), \quad \text{si } t \in [a + j\Delta, a + (j+1)\Delta] \quad (12.46)$$

Esta es la función continua y afín a trozos que toma los valores x_j en los puntos nodales $a + j\Delta$. Por tanto, se tiene

$$T(u) = \sum_{j=0}^{n} \int_{a+j\Delta}^{a+(j+1)\Delta} F\left(t, u(t), \frac{(x_{j+1} - x_j)}{\Delta}\right) dt$$

donde $u(t)$ en el intervalo de integración viene dada por (12.46). Teniendo en cuenta que u viene determinada por el vector $x = (x_1, \ldots, x_n)$ y que $T(u)$ se interpreta como una función de x, se está ante un problema de programación no lineal sin restricciones. Al resolverlo, se obtiene una solución aproximada de la solución del problema de optimización continuo del que se parte.

La estructura del funcional $T(u)$ en función de x depende de cada caso en particular. A continuación, se plantean y resuelven dos ejemplos mediante este método.

La braquistocrona

Este problema alcanzó la popularidad en el siglo pasado y fue uno de los primeros ejemplos para los que se encontró la solución óptima de manera explícita. Supóngase que se tienen dos puntos, A y B, a diferente altura en un plano. Se trata de encontrar el camino que une esos puntos de forma que una masa unitaria, bajo la acción gravitatoria y sin deslizamiento, emplee el menor tiempo posible en ir desde el punto más alto al más bajo. Este es uno de los ejemplos más conocidos entre los denominados problemas del camino más corto.

El primer paso consiste en derivar el funcional determinando el tiempo que tarda la masa en ir desde A hasta B, bajo la acción gravitatoria. Para mayor comodidad, se considera que el eje X se sitúa a lo largo de la dirección vertical de forma que la gravedad actúa a lo largo de este eje, y el eje Y se sitúa horizontalmente. Sean, sin pérdida de generalidad, $A = (1, 0)$ y $B = (0, 1)$. Supóngase que $y = u(x)$ es una curva continua que une A y B, de forma que $u(0) = 1$ y $u(1) = 0$. Si se supone que la masa unitaria viaja desde A hasta B a través del camino determinado mediante el grafo de u, ¿cuánto tardaría en ir desde A hasta B? Como el espacio es igual a la velocidad por el tiempo, se tiene que, en condiciones de continuidad, el espacio se puede medir a partir de la longitud de los elementos

$$d\ell = \sqrt{1+u'(x)^2}\,dx$$

mientras que la velocidad, en función de g, viene dada por

$$v = \sqrt{2gx}$$

Por tanto, si $T(u)$ representa el tiempo empleado en recorrer el camino $y = u(x)$, se tiene que

$$T(u) = \int_0^1 \frac{\sqrt{1+u'(x)^2}}{\sqrt{2gx}}\,dx$$

o análogamente, sabiendo que las constantes positivas que multiplican no afectan a la solución de los problemas de optimización, se tiene que

$$T(u) = \int_0^1 \frac{\sqrt{1+u'(x)^2}}{\sqrt{x}}\,dx$$

Este es el funcional a minimizar respecto a todas las funciones continuas $y = u(x)$ que satisfacen las restricciones $u(0) = 1$ y $u(1) = 0$.

Para comprender debidamente lo expuesto, se debe establecer la convexidad de los integrandos para T y la correspondiente ecuación de Euler–Lagrange. Puesto que esto es una tarea compleja, es mejor centrarse en el problema de optimización original y enfatizar cómo se pueden calcular mediante GAMS las soluciones óptimas para su versión discretizada.

La función objetivo discretizada, $\bar{T}(x)$, se puede expresar en función de los puntos nodales como

$$\bar{T}(x) = \sum_{j=0}^{n} \sqrt{1+\left(\frac{x_{j+1}-x_j}{\Delta}\right)^2} \int_{j/(n+1)}^{(j+1)/(n+1)} \frac{dx}{\sqrt{x}}$$

o incluso de manera más explícita, ignorando las constantes positivas

$$\bar{T}(x) = \sum_{j=0}^{n} \left(\sqrt{\frac{j+1}{n+1}} - \sqrt{\frac{j}{n+1}}\right) \sqrt{1+\left(\frac{x_{j+1}-x_j}{\Delta}\right)^2}$$

con $x_0 = 1$ y $x_{n+1} = 0$. Esta es la función que se minimiza, variando el número de subintervalos n. De esta forma, se llega a una solución que describe el arco de un cicloide, que es la curva óptima para el problema de optimización continuo.

Se muestra a continuación el código GAMS para este problema junto con los resultados.

```
$title FUNCIONAL1 n=20
SET J /0*20/;
VARIABLES z,x(J);
  x.fx(J)$(ord(J) eq 1) = 0;
  x.fx(J)$(ord(J) eq card(J)) = 1;
SCALAR n;
```

```
  n = card(J)-2;
EQUATION
  cost  funcion objetivo ;
cost.. z =e= SUM(J$(ord(J) lt card(J)),
(sqrt(ord(J))-sqrt(ord(J)-1))*sqrt(1+sqr(n+1)*sqr(x(J+1)-x(J))));
MODEL funct1 /all/;
SOLVE funct1 USING nlp MINIMIZING z;

                      LOWER     LEVEL     UPPER    MARGINAL
---- VAR Z             -INF      5.776     +INF       .
---- VAR X
     LOWER     LEVEL     UPPER    MARGINAL
0      .         .         .       -2.091
1     -INF      0.005     +INF  -1.163E-6
2     -INF      0.018     +INF  1.6563E-6
3     -INF      0.036     +INF  8.1950E-7
4     -INF      0.057     +INF  -1.405E-6
5     -INF      0.082     +INF  -4.992E-7
6     -INF      0.110     +INF  -1.353E-6
7     -INF      0.141     +INF  -3.233E-6
8     -INF      0.176     +INF  3.1623E-7
9     -INF      0.215     +INF  3.0808E-6
10    -INF      0.257     +INF  1.8503E-6
11    -INF      0.303     +INF  -6.250E-7
12    -INF      0.353     +INF  -1.116E-6
13    -INF      0.408     +INF  3.2812E-6
14    -INF      0.468     +INF  -1.583E-6
15    -INF      0.534     +INF       EPS
16    -INF      0.606     +INF  7.5934E-7
17    -INF      0.687     +INF  -8.374E-7
18    -INF      0.777     +INF       EPS
19    -INF      0.880     +INF       EPS
20    1.000     1.000     1.000     2.091
```

Al superponer los valores calculados sobre los correspondientes a la solución exacta, es imposible distinguir entre unos y otros (véase la figura 12.5).

Alternativamente, se pueden optar por un esquema de discretización que considere las pendientes en cada intervalo como variables independientes. Con este enfoque, la estructura de la función objetivo es más sencilla, pero es necesario incluir una restricción (lineal) para que las pendientes de los distintos intervalos sean tales que se conozca el valor de u en 1, y esto restringe el conjunto de posibles pendientes. En lugar de resolver el ejemplo anterior mediante este nuevo esquema con una restricción integral, se presenta un nuevo ejemplo que lo ilustra.

La cuerda colgante

El problema de la cuerda colgante es también un problema clásico planteado por Euler, que se ha resuelto mediante técnicas variacionales. Este problema consiste en determinar la forma que adopta una cuerda que cuelga libremente sobre sus extremos de igual longitud, bajo la acción de su propio peso (véase la figura 12.6).

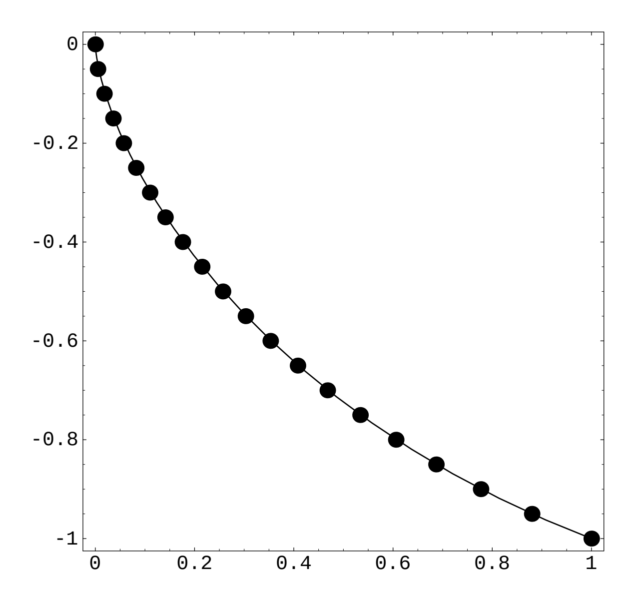


Figura 12.5: Soluciones exactas (línea continua) y aproximadas (línea de puntos) del problema de la braquistocrona.

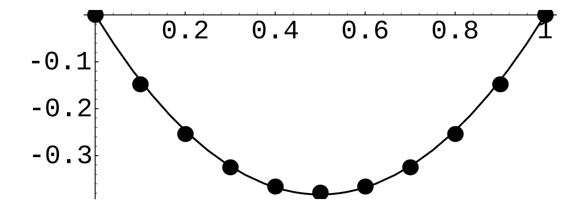


Figura 12.6: Soluciones exacta (línea continua) y aproximada (línea de puntos) al problema de la cuerda colgante.

Es claro que la forma óptima debe ser la que se corresponde con la menor energía potencial. Se supone que el cable tiene una sección transversal constante a lo largo de su longitud. Sea H la distancia entre los extremos de la cuerda, y sea L la longitud de la cuerda que se haya en el mismo plano horizontal que los extremos. Lógicamente, debe cumplirse que $L > H$ para que el problema tenga sentido. Supóngase que los extremos de la cuerda se disponen a lo largo del eje X, y que el eje Y comienza verticalmente en el extremo izquierdo de la cuerda. De esta forma, $(0, 0)$ y $(H, 0)$ son las coordenadas de los dos extremos fijos. Sea $y = u(x)$, $x \in (0, H)$ una función continua que une los dos extremos. Si la cuerda adopta el perfil descrito mediante el grafo de u, su energía potencial

se mide con la siguiente expresión

$$k \int_0^H u(x)\sqrt{1+u'(x)^2}\,dx$$

donde $k > 0$ es una constante relacionada con el material con el que está hecha la cuerda, $u(x)$ es la altura de cada sección transversal del material, y

$$\sqrt{1+u'(x)^2}\,dx$$

es la longitud de un elemento. Por tanto, una vez eliminadas las constantes positivas, se busca el perfil óptimo $u(x)$ de forma que se minimice el funcional de energía potencial

$$P(u) = \int_0^H u(x)\sqrt{1+u'(x)^2}\,dx$$

sujeto a $u(0) = 0$, $u(H) = 0$. En este caso, sin embargo, se necesita una restricción adicional para obligar a que L sea la longitud total de la cuerda. Si no, el problema de optimización no tendría sentido puesto que la longitud de la cuerda aumentaría indefinidamente para que la energía potencial fuese $-\infty$. Para evitarlo, se debe incluir la siguiente restricción

$$\int_0^H \sqrt{1+u'(x)^2}\,dx = L$$

Con lo que el problema de optimización consiste en minimizar

$$P(u) = \int_0^H u(x)\sqrt{1+u'(x)^2}\,dx$$

sujeto a

$$\int_0^H \sqrt{1+u'(x)^2}\,dx = L \quad u(0) = 0, \;\; u(H) = 0$$

Una vez más, para realizar un análisis exhaustivo de este problema se requiere el empleo de los multiplicadores de Lagrange asociados a la restricción de integración, y el estudio de la ecuación Euler–Lagrange para el Lagrangiano aumentado. Alternativamente, es posible obtener una versión discretizada de este problema y resolverlo mediante GAMS, como se hizo con anterioridad.

El proceso es el mismo que antes. Se restringe el problema de optimización al subespacio de dimensión finita de las funciones afines a trozos, que vienen determinadas por los valores nodales x_j, $j = 1, 2, \ldots, n$, para una partición uniforme del intervalo $(0, H)$. Sin pérdida de generalidad, se puede considerar $H = 1$. Como se hizo anteriormente, se tiene

$$P(x) = \sum_{j=0}^{n} \int_{j/(n+1)}^{(j+1)/(n+1)} u(x)\sqrt{1+\left(\frac{x_{j+1}-x_j}{\Delta}\right)^2}\,dx$$

donde de nuevo $\Delta = 1/(n+1)$. Dado que las raíces cuadradas bajo la integral son constantes, lo anterior se puede reescribir como

$$P(x) = \sum_{j=0}^{n} \sqrt{1 + \left(\frac{x_{j+1} - x_j}{\Delta}\right)^2} \int_{j/(n+1)}^{(j+1)/(n+1)} u(x)\, dx$$

Además, es posible emplear la regla trapezoidal para obtener una buena aproximación de la integral

$$\int_{j/(n+1)}^{(j+1)/(n+1)} u(x)\, dx \simeq \frac{x_{j+1} + x_j}{2(n+1)}$$

Con todo lo anterior, se tiene la siguiente función objetivo

$$P(x) = \sum_{j=0}^{n} \sqrt{1 + \left(\frac{x_{j+1} - x_j}{\Delta}\right)^2} \frac{x_{j+1} + x_j}{2(n+1)}$$

Finalmente, al expresar la restricción integral inicial en función del vector x, se tiene

$$L = \frac{1}{n+1} \sum_{j=0}^{n} \sqrt{1 + \left(\frac{x_{j+1} - x_j}{\Delta}\right)^2}$$

donde $x_0 = 0$ y $x_{n+1} = H$. Esta formulación discretizada y de dimensión finita es adecuada para la utilización de GAMS. Gracias a esta herramienta es posible encontrar buenas aproximaciones del problema clásico de la catenaria, que es la solución óptima del problema continuo.

A continuación, se muestra el código GAMS que se emplea para aproximar la solución óptima y los resultados obtenidos. Se toma L igual a 1.3116.

```
$title FUNCIONAL2 n=10
SET J /0*10/;
VARIABLES z,x(J);
  x.fx(J)$(ord(J) eq 1) = 0;
  x.fx(J)$(ord(J) eq card(J)) = 0;
  x.l(J)=-0.5;
SCALAR n;
  n = card(J)-2;
EQUATION
  cost   funcion objetivo
  rest   restriccion de igualdad;
cost.. z =e= SUM(J$(ord(J) lt card(J)),
                sqrt(1+sqr(n+1)*sqr(x(J+1)-x(J)))*(x(J+1)+x(J)));
rest.. (n+1)*1.3116 =e= SUM(J$(ord(J) lt
card(J)),sqrt(1+sqr(n+1)*sqr(x(J+1)-x(J))));
MODEL func2 /all/;
SOLVE func2 USING nlp MINIMIZING z;
```

```
                       LOWER     LEVEL     UPPER    MARGINAL
---- VAR Z              -INF     -6.114     +INF        .
---- VAR X
```

	LOWER	LEVEL	UPPER	MARGINAL
0	.	.	.	13.116
1	-INF	-0.148	+INF	EPS
2	-INF	-0.254	+INF	6.1259E-7
3	-INF	-0.325	+INF	8.3847E-7
4	-INF	-0.366	+INF	6.0564E-7
5	-INF	-0.379	+INF	.
6	-INF	-0.366	+INF	1.2127E-6
7	-INF	-0.325	+INF	1.0593E-6
8	-INF	-0.254	+INF	-5.058E-7
9	-INF	-0.148	+INF	8.8932E-7
10	.	.	.	13.116

La figura 12.6 muestra un gráfico con las soluciones exacta y aproximada.

12.5.2 Problemas de control óptimo

Los problemas de control óptimo son problemas de optimización continuos más complejos que sus correspondientes variacionales. De hecho, los problemas variacionales son un subconjunto de los problemas de control óptimo. Las principales componentes que presenta un problema de control óptimo son:

1. Un vector $x(t)$ que determina el comportamiento dinámico del estado de un sistema bajo control; el número de elementos de x indica el número de parámetros a determinar para identificar el estado del sistema considerado.

2. Un vector $u(t)$ que designa el control que puede ejercerse sobre el sistema de forma que se modifique su comportamiento dinámico teniendo en cuenta el objetivo perseguido; en muchas ocasiones, el control se limita al imponer que $u(t) \in K$ para un K dado.

3. La ecuación de estado

$$x'(t) = f(t, x(t), u(t)), \quad t \in (0, T)$$

   determina el comportamiento dinámico del sistema y expresa la interacción existente entre los estados y los controles.

4. Restricciones adicionales sobre el estado inicial y/o final del sistema.

5. El funcional objetivo

$$I(u) = \int_0^T g(t, x(t), u(t))\, dt$$

   que ofrece una medida de la optimalidad cuando actúa el control $u(t)$ en el sistema con un comportamiento dinámico resultante $x(t)$, que se obtiene al resolver la ecuación de estado junto con las condiciones inicial y final.

El objetivo del problema del control óptimo es encontrar la mejor forma de actuar sobre el sistema, el control óptimo $u(t)$, medido en términos del funcional

de coste propuesto, $I(u)$. La relación entre el estado x y el control u, establecida mediante la ecuación de estado y las restricciones adicionales, es lo que complica la resolución de los problemas de control óptimo respecto a los problemas variacionales. Obsérvese que los problemas variacionales se corresponden con la ecuación de estado

$$x'(t) = u(t),$$

donde los estados inicial y final se establecen a priori.

El proceso de cálculo es, asimismo, muy complejo puesto que requiere la resolución (normalmente numérica) de la ecuación de estado. No se pretende analizar este proceso en detalle; el objetivo es mostrar cómo algunos ejemplos de control óptimo se pueden resolver mediante GAMS, junto a alguna herramienta adicional que permita resolver la ecuación de estado, para aproximarse al control óptimo.

Control óptimo para el alcance de un blanco

Supóngase que se quiere alcanzar un blanco que está a una distancia $3 + \frac{5}{6}$ unidades de longitud en 3 unidades de tiempo. Entonces, el control que se puede ejercer sobre el proyectil es el módulo de la aceleración $u(t)$, de forma que el estado del proyectil $(x(t), x'(t))$ debe satisfacer la ecuación de estado y las condiciones auxiliares

$$x''(t) = u(t), \quad x(0) = x'(0) = 0, \quad x(3) = 3 + \frac{5}{6}$$

Las condiciones iniciales representan el hecho de que el proyectil parte del reposo. Además, se debe satisfacer la restricción que limita el tamaño de $u(t)$ haciendo que $0 \leq u \leq 1$. Finalmente, el objetivo consiste en alcanzar el blanco de la manera más económica, y esto se mide mediante el funcional de coste

$$I(u) = k \int_0^3 u(t)^2 \, dt$$

donde $k > 0$ es una constante.

Este es un típico problema de control óptimo. La estructura particular de los datos permite obtener analíticamente la solución óptima, aplicando las condiciones de optimalidad que plantea el principio de Pontryagin's (véase, por ejemplo, Polak [86] o Troutman [99]). De hecho, el control óptimo (único) para este problema es

$$u(t) = \begin{cases} 1, & t \leq 1 \\ \dfrac{3-t}{2}, & t \geq 1 \end{cases}$$

La justificación de lo anterior escapa los objetivos de este libro. Sin embargo, es posible comparar la solución exacta y la aproximada calculada mediante GAMS.

La obtención de una versión discretizada para problemas de control óptimo es, como se dijo anteriormente, una tarea compleja puesto que se necesita resolver la ecuación de estado. Para este ejemplo concreto, se puede conseguir una

versión explícita que puede programarse mediante GAMS. Sin embargo, la idea básica que subyace en la discretización es la misma: se divide el intervalo $(0,3)$ en n subintervalos iguales y se hace que el control u tome el valor constante u_j en el subintervalo asociado $(3(j-1)/n, 3j/n)$. De esta forma, una vez resuelta la ecuación de estado para este tipo de controles constantes a trozos, se puede expresar el problema de control discretizado como un problema de programación matemática (no lineal y con restricciones) en las variables u_j.

En el subintervalo $(3(j-1)/n, 3j/n)$, el control u toma el valor constante u_j; por tanto, suponiendo recursivamente que se ha resuelto la ecuación de estado en el subintervalo anterior $(3(j-2)/n, 3(j-1)/n)$ y tomando a_{j-1} y b_{j-1} como valores de la velocidad y situación, respectivamente, para $t = 3(j-1)/n$, se necesita encontrar la solución de

$$x''(t) = u_j, \quad x'\left(\frac{3(j-1)}{n}\right) = a_{j-1}, \; x\left(\frac{3(j-1)}{n}\right) = b_{j-1}$$

para obtener

$$x(t) = \frac{u_j}{2}\left(t - \frac{3(j-1)}{n}\right)^2 + a_{j-1}\left(t - \frac{3(j-1)}{n}\right) + b_{j-1}$$

Los valores de velocidad y situación para $t = 3j/n$ son

$$a_j = a_{j-1} + \frac{3u_j}{2n}, \quad b_j = b_{j-1} + a_{j-1}\frac{3}{n} + \frac{9u_j}{2n^2}$$

Obsérvese que las condiciones iniciales implican que $a_0 = b_0 = 0$. Aplicando las fórmulas recursivas iterativamente, se puede deducir que

$$a_j = \frac{3}{n}\sum_{k=1}^{j} u_k, \quad b_j = \frac{9}{2n^2}\sum_{k=1}^{j}(2j-2k+1)u_k$$

La condición final $x(3) = 3 + \frac{5}{6}$ se transforma en la restricción lineal

$$\frac{9}{2n^2}\sum_{k=1}^{n}(2n-2k+1)u_k = 3 + \frac{5}{6}$$

Por otro lado, el funcional cuadrático de coste, si se ignoran las constantes positivas, se expresa como

$$I(u) = \sum_{k=1}^{n} u_k^2, \quad u = (u_k); \; k = 1, 2, \ldots, n$$

Por tanto, se debe resolver el problema de programación (cuadrática). Minimizar

$$Z = \sum_{k=1}^{n} u_k^2$$

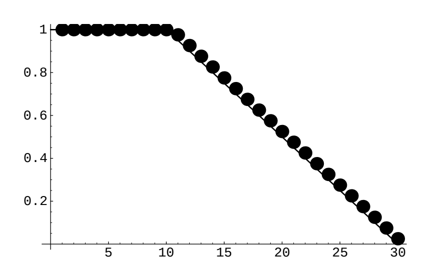


Figura 12.7: Soluciones exacta (línea continua) y aproximada (línea de puntos) del problema de control óptimo de alcance de un blanco.

sujeto a

$$\frac{9}{2n^2}\sum_{k=1}^{n}(2n-2k+1)u_k = 3+\frac{5}{6}, \quad 0 \leq u_k \leq 1$$

Al implementar este problema en GAMS se puede encontrar una solución al problema de control óptimo de partida, muy cercana a la solución exacta obtenida anteriormente. La figura 12.7 muestra las soluciones exacta y numérica de este problema. El código GAMS y las soluciones obtenidas se muestran a continuación.

```
$title FUNCIONAL3 n=30
SET K /1*30/;
VARIABLES z,u(K);
  u.up(K) = 1;
  u.lo(K) = 0;
SCALAR n;
  n = card(K);
EQUATION
  cost   funcion objetivo
  rest   restriccion de igualdad;
cost.. z =e= SUM(K,sqr(u(K)));
rest.. 3+(5/6) =e= (9/(2*sqr(n)))*SUM(K,(2*n-2*ord(K)+1)*u(K));
MODEL funct3 /all/;
SOLVE funct3 USING nlp MINIMIZING z;
```

```
                      LOWER     LEVEL     UPPER    MARGINAL
---- VAR Z             -INF     16.671     +INF       .
---- VAR U
      LOWER     LEVEL     UPPER    MARGINAL
1       .        1.000     1.000    -0.952
2       .        1.000     1.000    -0.852
3       .        1.000     1.000    -0.752
4       .        1.000     1.000    -0.652
5       .        1.000     1.000    -0.552
6       .        1.000     1.000    -0.452
7       .        1.000     1.000    -0.351
8       .        1.000     1.000    -0.251
9       .        1.000     1.000    -0.151
10      .        1.000     1.000    -0.051
11      .        0.976     1.000      EPS
12      .        0.926     1.000 3.4120E-6
```

```
13       .        0.876     1.000 2.7012E-6
14       .        0.826     1.000 1.9903E-6
15       .        0.775     1.000 1.2794E-6
16       .        0.725     1.000 -2.347E-7
17       .        0.675     1.000 -2.185E-7
18       .        0.625     1.000        .
19       .        0.575     1.000 -1.564E-6
20       .        0.525     1.000      EPS
21       .        0.475     1.000      EPS
22       .        0.425     1.000      EPS
23       .        0.375     1.000      EPS
24       .        0.325     1.000      EPS
25       .        0.275     1.000 -1.026E-5
26       .        0.225     1.000 -8.395E-6
27       .        0.175     1.000 -6.530E-6
28       .        0.125     1.000 -4.664E-6
29       .        0.075     1.000 -2.798E-6
30       .        0.025     1.000      EPS
```

Control óptimo de un oscilador armónico

Dentro de los problemas de control óptimo se encuentra una clase de problemas de especial importancia, cuyo objetivo es realizar una tarea determinada en el menor tiempo posible. Estos casos responden a la ecuación de estado siguiente

$$x'(t) = f(t, x(t), u(t)), \quad t \in (0, T), u(t) \in K$$

a la que se añaden las condiciones que deben cumplir los estados inicial y final x_I y x_F, respectivamente. El objetivo consiste en encontrar el control u para realizar la tarea escogida

$$x(T) = x_F$$

para el menor valor posible de T.

En concreto, se presenta el ejemplo de un oscilador armónico regido por la siguiente ecuación de estado

$$x''(t) + x(t) = u(t), \quad t \in (0, T), |u| \leq 1$$

El fin que se persigue es llevar al oscilador de unas condiciones iniciales conocidas

$$x(0) = a_0, \;\; x'(0) = b_0$$

al reposo

$$x(T) = x'(T) = 0$$

en el menor tiempo posible. De nuevo, puede calcularse la solución del problema analíticamente mediante la aplicación de las condiciones de optimalidad del principio de Pontryaguin. En concreto, la dependencia lineal (de hecho, constante) del funcional de coste respecto al control implica que sólo se ejercerá el control en los valores extremos $+1$ y -1. Conocida la solución exacta del problema, se procede, a continuación, a obtener una solución aproximada, resolviendo la versión discretizada del problema mediante GAMS

Puesto que en este caso el tiempo T es desconocido, se debe considerar en el programa como una variable independiente del problema, de forma que $u = (u_j)_{j=0,1,\dots,n}$ son las variables, donde u_0 representa a T y el correspondiente control constante a trozos toma el valor u_j en el intervalo $(u_0(j-1)/n, u_0 j/n)$. Igual que antes, si se considera

$$a_j = x\left(\frac{u_0 j}{n}\right), \quad b_j = x'\left(\frac{u_0 j}{n}\right)$$

entonces es posible determinar x mediante resolución iterativa

$$x''(t) + x(t) = u_j, \quad x\left(\frac{u_0(j-1)}{n}\right) = a_{j-1}, x'\left(\frac{u_0(j-1)}{n}\right) = b_{j-1}$$

Para este ejemplo, resulta complicado establecer expresiones exactas para a_j y b_j. Sin embargo, se puede utilizar alguna herramienta de cálculo simbólico para encontrar estas expresiones. En cualquier caso, para obtener la solución aproximada de este problema no es necesario conocer estas expresiones exactas, basta con emplear un integrador de Euler cuyo paso de avance sea u_0/n para encontrar unas aproximaciones razonables de a_j y b_j. La notación matricial de las expresiones para el cálculo de a_j y b_j (sin distinguir entre valores exactos o aproximados) es como sigue

$$\begin{pmatrix} a_j \\ b_j \end{pmatrix} = \begin{pmatrix} a_{j-1} \\ b_{j-1} \end{pmatrix} + \frac{u_0}{n}\left[\begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}\begin{pmatrix} a_{j-1} \\ b_{j-1} \end{pmatrix} + \begin{pmatrix} 0 \\ u_j \end{pmatrix}\right]$$

para $j = 1, 2, \dots, n$. Si se emplea esta igualdad iterativamente, se obtiene

$$\begin{pmatrix} a_j \\ b_j \end{pmatrix} = \begin{pmatrix} 1 & \frac{u_0}{n} \\ -\frac{u_0}{n} & 1 \end{pmatrix}^j \begin{pmatrix} a_0 \\ b_0 \end{pmatrix} + \frac{u_0}{n}\sum_{k=0}^{j-1}\begin{pmatrix} 1 & \frac{u_0}{n} \\ -\frac{u_0}{n} & 1 \end{pmatrix}^k \begin{pmatrix} 0 \\ u_{j-k} \end{pmatrix}$$

Las restricciones del problema establecen el cumplimiento de las condiciones elegidas para el estado final de reposo $a_n = b_n = 0$. Por tanto, el problema de programación lineal discreto (no lineal y restringido) cuya solución óptima es una aproximación razonable de la solución del problema de partida consiste en minimizar

$$Z = u_0$$

sujeto a

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 & \frac{u_0}{n} \\ -\frac{u_0}{n} & 1 \end{pmatrix}^n \begin{pmatrix} a_0 \\ b_0 \end{pmatrix} + \frac{u_0}{n}\sum_{j=1}^{n}\begin{pmatrix} 1 & \frac{u_0}{n} \\ -\frac{u_0}{n} & 1 \end{pmatrix}^{n-j} \begin{pmatrix} 0 \\ u_j \end{pmatrix}$$

y

$$u_0 \geq 0, \quad -1 \leq u_j \leq 1, j = 1, 2, \dots, n$$

Tomando como datos conocidos las condiciones iniciales a_0 y b_0, y el valor de n, se puede obtener con GAMS una aproximación a la solución del problema del control óptimo de un oscilador armónico lineal.

Para simplificar la formulación en GAMS, se realiza el siguiente cambio de variables

$$\frac{u_0}{n} = \tan\alpha, \quad 0 \leq \alpha \leq \frac{\pi}{2}$$

de forma que

$$\begin{pmatrix} 1 & \frac{u_0}{n} \\ -\frac{u_0}{n} & 1 \end{pmatrix}^k = (1+\tan^2\alpha)^k \begin{pmatrix} \cos(k\alpha) & \sin(k\alpha) \\ -\sin(k\alpha) & \cos(k\alpha) \end{pmatrix}$$

Se ha resuelto el problema para distintos valores de las condiciones iniciales (a_0, b_0), como son: $(-1, 0)$, $(3, 0)$ y $(-7, 0)$. Se puede apreciar, teniendo en cuenta las posibles imprecisiones, el típico comportamiento oscilatorio (*bang–bang*) del control que va desde -1 hasta 1, como predice el principio de Pontryaguin.

El fichero de entrada GAMS, así como un extracto del fichero de salida para las condiciones iniciales $(a_0, b_0) = (3, 0)$ se presentan a continuación:

```
$title FUNCIONAL n=20, azero
SET J /1*20/;
SCALAR pi El numero pi /3.1416/;
VARIABLES z,theta,u(J);
 u.lo(J)=-1;
 u.up(J)=1;
 theta.lo=0;
 theta.up=pi/2;
 theta.l=pi/4;
 u.l(J)=0.;
SCALARS n, azero;
  n = card(J);
  alpha=-7;
EQUATION
  cost   funcion objetivo
  const1
  const2;
cost.. z =e= SIN(theta)/COS(theta);;
const1..
alpha*COS(n*theta)*POWER((1+POWER((SIN(theta)/COS(theta)),2)),n)+(SIN(theta)
/COS(theta))*
         SUM(J,u(J)*SIN((n-ORD(J))*theta)*
           POWER((1+POWER((SIN(theta)/COS(theta)),2)),n-ORD(J))
         ) =E= 0;
const2..
-azero*SIN(n*theta)*POWER((1+POWER((SIN(theta)/COS(theta)),2)),n)+(SIN(theta
)/COS(theta))*
         SUM(J,u(J)*COS((n-ORD(J))*theta)*
         POWER((1+POWER((SIN(theta)/COS(theta)),2)),n-ORD(J))
         ) =E= 0;
MODEL funct5 /all/;
SOLVE funct5 USING nlp MINIMIZING z;

  azero=3;
                       LOWER     LEVEL     UPPER    MARGINAL
---- VAR Z              -INF      0.982     +INF        .
---- VAR THETA           .        0.776     1.571       .
---- VAR U
      LOWER     LEVEL     UPPER    MARGINAL
```

1	-1.000	-1.000	1.000	1.2962E+5
2	-1.000	1.000	1.000	-2.455E+5
3	-1.000	1.000	1.000	-2.118E+5
4	-1.000	1.000	1.000	-9.025E+4
5	-1.000	1.000	1.000	-1.067E+4
6	-1.000	-1.000	1.000	15633.593
7	-1.000	-1.000	1.000	14116.404
8	-1.000	-1.000	1.000	6201.667
9	-1.000	-1.000	1.000	846.919
10	-1.000	1.000	1.000	-991.476
11	-1.000	1.000	1.000	-939.412
12	-1.000	1.000	1.000	-425.373
13	-1.000	1.000	1.000	-65.560
14	-1.000	-1.000	1.000	62.581
15	-1.000	-1.000	1.000	62.430
16	-1.000	-1.000	1.000	29.126
17	-1.000	-1.000	1.000	4.979
18	-1.000	1.000	1.000	-3.929
19	-1.000	1.000	1.000	-4.143
20	-1.000	1.000	1.000	-1.991

12.6 Sistemas de transporte

En este apartado se estudian algunos modelos de equilibrio en redes de transporte urbano. Se persigue un doble objetivo: (1) presentar algunos de estos modelos que se emplean en la práctica, y (2) ilustrar la versatilidad de GAMS para su implementación.

Puesto que no se analizan los algoritmos específicos para la resolución de este tipo de problemas, el lector interesado en esta materia puede consultar el libro de Patriksson [84]. Aquellos lectores que estén interesados en los modelos matemáticos aplicados en la planificación del transporte pueden consultar los excelentes libros de Ortúzar y Willumsen [82], y Sheffi [98]. El libro de Potts y Oliver [88] ofrece una magnífica introducción a estos problemas.

12.6.1 Introducción

El crecimiento económico ha originado un importante incremento de la demanda de transporte, tanto en los países desarrollados como en los que están en vías de desarrollo. Este aumento ha conducido a que, en algunas regiones y para ciertos modos de transporte, la demanda supere a la oferta de servicios de transporte, originando que viejos problemas como son la congestión, polución, accidentes, déficit financieros, etc. reaparezcan con nuevas apariencias y dificultades.

La demanda de servicios de transporte tiene una naturaleza dinámica, ésta varía durante cada momento del día, de un día a otro de la semana e incluso por meses. La oferta de servicios de transporte es altamente compleja, por un lado una autoridad provee la infraestructura y por otro lado los operadores (pueden existir varios operadores por cada modo de transporte) proveen los servicios de transporte. Ambos elementos, demanda y oferta de servicios de transporte, interactúan, originando una situación de equilibrio. La tarea de la planificación

del transporte urbano es predecir y controlar la evolución de estos puntos de equilibrio en el tiempo, de modo que el bienestar de la sociedad sea maximizado. La modelación de estos puntos de equilibrio permite entender mejor la evolución de éstos y ayuda al desarrollo e implementación del conjunto de estrategias de gestión y a los programas de inversión.

La modelación de esta situación de equilibrio ha motivado multitud de modelos matemáticos. La práctica desde la década de los años sesenta ha consolidado el esquema de aplicación llamado *modelo de cuatro etapas*:

1. *Fase de generación de viajes.* El esquema empieza considerando una zonificación del área de estudio, una codificación de la red de transporte y una base de datos para el estudio. Estos datos están referidos al nivel de actividad económica y demográfica de cada zona, que incluye el nivel de empleo, localización de centros comerciales, zonas recreativas, centros escolares, etc., y son empleados para estimar el número de viajes generados y atraídos por cada zona considerada en el estudio.

   Tras esta fase se obtiene un modelo de la red de transporte mediante un grafo $\mathcal{G} = (\mathcal{N}, \mathcal{A})$ donde $\mathcal{A}$ y $\mathcal{N}$ son el conjunto de arcos (dirigidos) y nudos respectivamente. El significado de los arcos depende de si la red es de tráfico o de transporte público. En el primer caso los arcos están asociados a las calles y los nudos a las intersecciones. En el segundo caso cada nudo está asociado a una parada y cada arco representa los posibles desplazamientos entre paradas que un usuario puede efectuar.

2. *Fase de distribución.* En esta fase se obtiene la distribución de los viajes sobre el espacio, esto es, se obtiene el número de viajes que se efectúan de una zona a otra, obteniéndose la denominada matriz de viajes origen-destino (O-D). En esta fase se determina un conjunto de pares ordenados de $\mathcal{N} \times \mathcal{N}$ y la demanda de viajes en ellos (que inicialmente se considera fija). Este conjunto de pares de demandas se denota por W y cada par O-D por $\Omega = (i, j)$, donde i es el origen y j es el destino. La demanda total de viajes para el par Ω se denota por $\bar{g}^{\Omega}$.

3. *Fase de partición modal.* En esta fase se obtiene una matriz O-D para cada modo de transporte presente en el estudio.

4. *Fase de asignación.* Finalmente, cada matriz de demanda O-D es asignada a un conjunto de rutas en la red de transporte. Usualmente se efectúa una asignación de tráfico por un lado (vehículos privados), y una asignación a la red de transporte público, por otro.

En la actualidad este esquema secuencial ha sido superado por métodos que integran dos o varias de estas etapas simultáneamente. No obstante, este esquema sigue siendo de utilidad a la hora de describir modelos o de comparar modelos alternativos.

En esta sección, se presentan cuatro modelos de equilibrio para la asignación de tráfico en redes congestionadas que han sido elaborados sobre la base

del llamado *primer principio de Wardrop* [102]. En la siguiente subsección se introducen los elementos necesarios para la formulación de estos modelos y posteriormente se desarrollan éstos en las subsecciones siguientes.

12.6.2 Elementos de una red de tráfico

Este apartado describe brevemente los principales elementos para la modelación de las redes de tráfico. Las redes de tráfico se representan mediante un modelo matemático denominado *grafo dirigido*. Un grafo dirigido se define como un par $\mathcal{G} = (\mathcal{N}, \mathcal{A})$, donde $\mathcal{N}$ es un conjunto finito de nudos, y $\mathcal{A}$ es un conjunto de pares ordenados (*arcos*) de elementos de $\mathcal{N}$. Los nudos se denotan mediante i o j, y los arcos mediante (i, j).

Los nudos de $\mathcal{N}$ se clasifican en dos categorías: *centroides* o *nudos intermedios*. El primer grupo representa cualquier zona donde se inicia un viaje (*origen*), o la zona donde se finaliza (*destino*). Los nudos intermedios representan las intersecciones de las calles.

Los arcos del grafo representan las vías (calles, circunvalaciones, etc.) de la red de transporte. Nótese que los pares (i, j) y (j, i) representan sentidos opuestos de una misma vía.

Ejemplo 12.8 (red de Nguyen–Dupuis). Sea el grafo $\mathcal{G}$ definido en la tabla 12.7, donde $\mathcal{N} = \{1, \ldots, 13\}$ y $\mathcal{A}$ tiene 19 arcos. Este ejemplo proviene de Nguyen y Dupuis [80] (red ND) y se ilustra en la figura 12.8.

Tabla 12.7: Arcos del grafo $\mathcal{G}$

Arcos			
(1, 5)	(1, 12)	(4, 5)	(4, 9)
(5, 6)	(5, 9)	(6, 7)	(6, 10)
(7, 8)	(7, 11)	(8, 2)	(9, 10)
(9, 13)	(10, 11)	(11, 2)	(11, 3)
(12, 6)	(12, 8)	(13, 3)	

■

El problema de flujos en redes con un único producto

Muchos problemas de planificación se representan mediante un grafo, como el ejemplo anteriormente mencionado del tráfico en una ciudad. Una característica esencial de estos problemas es que se producen flujos de vehículos, de mercancías, de pasajeros, etc. de una parte del grafo a otra. En este caso, el grafo se denomina *red* o, concretamente, si se quiere enfatizar que se trata de una aplicación de transporte, se denomina *red de transporte*.

Los problemas de flujos en redes se clasifican en: problemas de un sólo producto y problemas de múltiples productos. En el primer caso todos los flujos

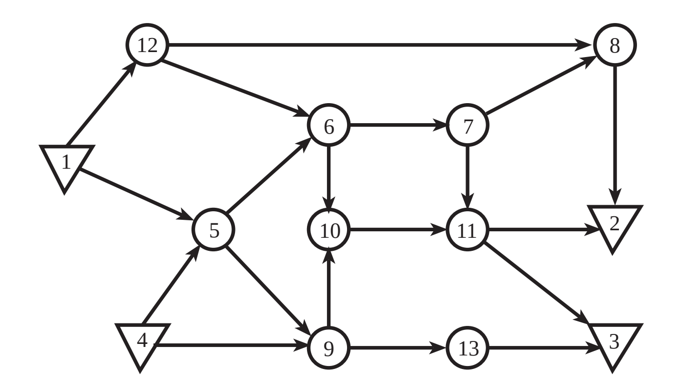


Figura 12.8: Representación de una red de tráfico mediante un grafo. Los círculos representan nudos intermedios y los triángulos son los orígenes o los destinos.

de la red tienen la misma naturaleza y en el segundo caso existen diferencias entre los flujos, por ejemplo problemas con dos mercancías diferentes.

Cualquier red con un sólo producto, bien sea eléctrica, de distribución de agua, o de transporte, debe satisfacer la *ley de la conservación del flujo*, que garantiza que la suma algebraica de los flujos que entran en un nudo menos los que salen de él es igual al flujo que se ha consumido y/o generado en dicho nudo. Este tipo de condiciones se denominan *restricciones de red*.

Estos problemas asumen que existe una función (en general, no lineal) $C_{ij}(f_{ij})$ para cada arco (i, j) de la red que determina el coste ocasionado por atravesar f_{ij} unidades de flujo el citado arco. El objetivo de estos problemas es minimizar la suma de los costes generados en todos los arcos.

Las cuatro componentes principales de este problema son

1. **Datos.**

   $\mathcal{N}$**:** conjunto de nudos en la red

   $\mathcal{A}$**:** conjunto de arcos en la red

   r_i**:** flujo que se inicia o finaliza en el nudo i. Si i es un nudo intermedio entonces $r_i = 0$; si i es un nudo origen, $r_i > 0$; por último, si i es un nudo destino, $r_i < 0$

   $A(i)$**:** conjunto de nudos $\{j : \ j \in \mathcal{N}, (i, j) \in \mathcal{A}\}$ "después" del nudo i

   $B(i)$**:** conjunto de nudos $\{j : \ j \in \mathcal{N}, (j, i) \in \mathcal{A}\}$ "antes" del nudo i

   $C_{ij}(x)$**:** coste del arco $i - j$ para un flujo x

2. **Variables.**

   f_{ij}**:** flujo en el arco (i, j)

3. **Restricciones.** Las ecuaciones de conservación del flujo se formulan de la siguiente manera:

$$\sum_{j \in A(i)} f_{ij} - \sum_{j \in B(i)} f_{ji} = r_i, \ \forall i \in \mathcal{N}$$

4. **Función a optimizar.** En este problema, se minimiza

$$Z = \sum_{(i,j) \in \mathcal{A}} C_{ij}(f_{ij}) \qquad (12.47)$$

Ejemplo 12.9 (problema de un único producto). Para ilustrar el comportamiento de las ecuaciones de conservación del flujo, considérese la red de la figura 12.8, donde los círculos representan los nudos intermedios, y los triángulos representan los centroides (orígenes y destinos). En el nudo intermedio 5, se tiene que

$$i = 5, \quad A(5) = \{6, 9\}, \quad B(5) = \{1, 4\}$$

$$\sum_{j \in A(5)} f_{5j} - \sum_{j \in B(5)} f_{j5} = f_{5,6} + f_{5,9} - f_{1,5} - f_{4,5} = r_5 = 0$$

■

Problemas de flujos en redes con múltiples productos

Anteriormente, se han considerado aquellos problemas en los que los flujos son de un solo tipo. En este apartado se presenta un problema análogo al anterior pero se diferencia en que las entidades que circulan por la red tienen distinta naturaleza.

El problema de asignación de tráfico es un ejemplo de esta clase de problemas. En la red de tráfico existen distintos tipos de usuarios en función del origen y destino de su viaje. Si no se contemplase este hecho, podría ocurrir que un usuario terminase su viaje en un destino diferente al deseado, y al tiempo se cumplieran las ecuaciones de conservación de flujo para los orígenes, destinos y nudos intermedios. Esto obliga a que para cada *producto*, definido por un par O–D, se deben satisfacer sus propias ecuaciones de conservación de flujo. Se denota por $\Omega = (i, j)$ a un par concreto origen–destino (O–D), y mediante W al conjunto de todos los pares O–D.

Para formular el problema de asignación de tráfico, se supone que la demanda de viajes para el par Ω (entre el origen O^{Ω} y el destino D^{Ω}) es $g^{\Omega} > 0$. La red ND (figura 12.8) va a usarse para ilustrar los modelos de asignación en equilibrio. Esta red tiene cuatro pares O-D que se muestran en la tabla 12.8.

Por tanto, las restricciones del problema de múltiples productos se puede plantear como se muestra a continuación

Tabla 12.8: Pares O–D para la red ND

Par	Demanda g^Ω	Par	Demanda g^Ω
$\Omega_1 = (1,2)$	400	$\Omega_2 = (1,3)$	800
$\Omega_3 = (4,2)$	600	$\Omega_4 = (4,3)$	200

(12.48)

1. Ecuaciones de conservación del flujo para cada producto:

$$\sum_{j\in A(i)} f_{ij}^\Omega - \sum_{j\in B(i)} f_{ji}^\Omega = r_i^\Omega, \ \forall i \in \mathcal{N}, \quad \forall \Omega \in W$$

donde

$$r_i^\Omega = \begin{cases} g^\Omega & \text{si } O^\Omega = i \\ -g^\Omega & \text{si } D^\Omega = i \\ 0 & \text{si } i \text{ es un nudo intermedio} \\ & \text{para el producto } \Omega. \end{cases}$$

2. El flujo total en la red se obtiene superponiendo el flujo de todos los productos para cada arco:

$$\sum_{\Omega\in W} f_{ij}^\Omega = f_{ij}, \ \ \forall (i,j) \in \mathcal{A}$$

Las relaciones anteriores se pueden expresar en notación matricial como

$$\begin{array}{rcl} \mathbf{E}\mathbf{f}^\Omega & = & \mathbf{r}^\Omega \ \ \forall \Omega \in W \\ \sum\limits_{\Omega\in W} \mathbf{f}^\Omega & = & \mathbf{f} \end{array}$$

donde $\mathbf{E}$ es la matriz de dimensión $n \times l$ de incidencias nudo–arco, cuyo elemento correspondiente a la fila del nudo i y la columna del arco (j,k) se define como:

$$e_{i(jk)} = \begin{cases} +1 & \text{si } i = j \\ -1 & \text{si } i = k \\ 0 & \text{en otro caso} \end{cases}$$

Obsérvese que las ecuaciones de conservación del flujo equivalen a replicar la red tantas veces como productos diferentes se consideren. Una forma de reducir el número de estas réplicas consiste en emplear una sola red para todos aquellos pares O–D que tienen el mismo origen o, de manera alternativa, el mismo destino.

Tráfico congestionado

Hasta ahora se ha considerado la estructura de las redes de tráfico. En este apartado se considera el efecto de la congestión en los tiempos de viajes en la

red. El aumento de los volúmenes de tráfico en los arcos reduce la velocidad de transporte, produciendo un incremento en el tiempo de viaje. Para tener en cuenta este efecto, se introduce el concepto de *coste de viaje en el arco*, $c_{ij}(f_{ij})$, que representa el tiempo medio en recorrer el trayecto definido por el arco (i, j) para un flujo de f_{ij} unidades. Para el análisis del tráfico, estas funciones se consideran estrictamente crecientes, no lineales y positivas. Sus parámetros principales son el *tiempo de viaje con el arco vacío*, c_{ij}^0 y la capacidad real del arco, k_{ij}, que es una medida a partir de la cual el tiempo de viaje se incrementará rápidamente al incrementar el flujo. La expresión más empleada de $c_{ij}(f_{ij})$ se denomina función BPR

$$c_{ij}(f_{ij}) = c_{ij}^0 + b_{ij}(f_{ij}/k_{ij})^{n_{ij}}, \tag{12.49}$$

donde b_{ij} y n_{ij} son parámetros a ajustar experimentalmente.

12.6.3 El problema de asignación de tráfico

En este apartado se estudiarán modelos matemáticos que han sido planteados para resolver el problema de la asignación de tráfico a las rutas de la red y se clasifican en *dinámicos* o *estáticos*, (sólo se abordarán aquí modelos estáticos). Estos modelos se centran en unas pocas horas del día, como las horas puntas, y trabajan con valores medios (demandas, tiempos, flujos, etc.) durante el período de estudio.

Además se debe asumir un principio para la modelación de la elección que hacen los usuarios de la ruta en la red de transporte. Un marco para la elaboración de estos modelos, llamados modelos de *asignación en equilibrio*, lo constituye el *primer principio de Wardrop* [102] que se enuncia del siguiente modo:

> "En el equilibrio ningún usuario puede reducir el coste de su viaje mediante cambio de ruta."

Este principio implica que todos los tiempos de viaje empleados en todas las rutas usadas para satisfacer el mismo par O-D deben ser iguales y menores o iguales que el tiempo de viaje en cualquier otra ruta no empleada para satisfacer dicho par de demanda. Este principio ha sido empleado para construir modelos de equilibrio, tanto en redes de tráfico como en redes de transporte público.

El primer principio de Wardrop, también denominado DUE (*deterministic user equilibrium*), asume que todos los usuarios perciben el coste de la misma manera y además, conocen los costes de todas las rutas (tienen información perfecta). En la realidad las percepciones de los costes están sujetas a variaciones y los usuarios eligen la ruta de acuerdo a su percepción. Se han elaborado modelos de equilibrio en los que los costes de viajes son la suma de una parte fija más una componente aleatoria, en este caso los usuarios eligen una u otra ruta dependiendo de la distribución de probabilidad de los costes aleatorios. Esta forma de asignación recibe el nombre *asignación estocástica* (SUE).

Existe otro marco en la elaboración de modelos de equilibrio, el llamado *segundo principio de Wardrop* que asume que los usuarios pueden ser persuadidos a emplear cualquier ruta y por tanto, los usuarios serán asignados a las rutas que minimicen el tiempo total empleado por el sistema de transporte. Este principio se enuncia

> "Los usuarios eligen la ruta de modo que se minimice el tiempo total de transporte en la red."

El primer principio de Wardrop es utilizado para modelar el comportamiento de los usuarios, mientras que el segundo principio es usado como un criterio para diseñar la red de transporte. El primer principio asume que los usuarios actúan individualmente mientras que el segundo asume que los usuarios buscan el óptimo del sistema (de todos los usuarios)

Beckman y McGuire [10] formularon el siguiente problema que expresa la condición de equilibrio según el primer principio de Wardrop denominado *problema de asignación de tráfico* (PAT). Minimizar

$$Z = \sum_{(i,j)\in\mathcal{A}} \int_0^{f_{ij}} c_{ij}(x)dx \tag{12.50}$$

sujeto a

$$\begin{aligned} \sum_{j\in A(i)} f_{ij}^{\Omega} - \sum_{j\in B(i)} f_{ji}^{\Omega} &= r_i^{\Omega}, \ \forall i \in \mathcal{N}, \ \ \forall \Omega \in W \\ \sum_{\Omega\in W} f_{ij}^{\Omega} &= f_{ij}, \ \ \forall (i,j) \in \mathcal{A} \\ f_{ij}^{\Omega} &\geq 0, \quad \forall (i,j) \in \mathcal{A}, \forall \Omega \in W \end{aligned}$$

La formulación del problema anterior se conoce como la formulación *nudo-arco.* Como las condiciones de equilibrio vienen dadas en función de los costes y flujos de las rutas, el problema de optimización anterior ha de basarse en estos elementos. A continuación, se presenta una formulación alternativa de las condiciones de equilibrio, basada en los flujos en los caminos y en los arcos.

Formulación basada en los flujos en los caminos

Los componentes básicos de esta formulación son

1. **Datos.**

   $\mathcal{R}_\Omega$**:** conjunto de caminos asociados al producto Ω

   $c_a(x)$**:** coste asociado al flujo x a través del arco a

2. **Variables.**

   h_r**:** flujo por la ruta r

3. **Restricciones.** La cantidad de viajeros para un par Ω de demanda es la suma de la cantidad total de viajeros que siguiendo distintos caminos satisfacen dicha demanda

$$\sum_{r \in \mathcal{R}_\Omega} h_r = g_\Omega, \ \forall \Omega \in W \tag{12.51}$$

Además, el flujo debe ser no negativo

$$h_r \geq 0, \ \forall r \in \mathcal{R}_\Omega, \ \forall \Omega \in W \tag{12.52}$$

La relación entre el flujo de los arcos y el flujo de los caminos es que el flujo en cada arco $a \in \mathcal{A}$ es la suma del flujo de todos los caminos que emplean dicho arco:

$$\sum_{w \in W} \sum_{r \in \mathcal{R}_\Omega} \delta_{a,r} h_r = f_a \quad \forall a \in \mathcal{A} \tag{12.53}$$

donde

$$\delta_{a,r} = \begin{cases} 1 & \text{si } r \in \mathcal{R}_\Omega \text{ contiene al arco } a \\ 0 & \text{en otro caso} \end{cases}$$

4. **Función a optimizar.** En este problema, se minimiza

$$Z = \sum_{a \in \mathcal{A}} \int_0^{f_a} c_a(x) dx$$

Por tanto, la formulación en el espacio de flujos en los arcos del PAT se puede plantear como el siguiente problema de optimización. Minimizar

$$Z = \sum_{a \in \mathcal{A}} \int_0^{f_a} c_a(x) dx$$

sujeto a

$$\begin{aligned} \sum_{r \in \mathcal{R}_\Omega} h_r &= g_\Omega, \ \forall \Omega \in W & (12.54) \\ \sum_{\Omega \in W} \sum_{r \in \mathcal{R}_\Omega} \delta_{ar} h_r &= f_a \ , \forall a \in \mathcal{A} & (12.55) \\ h_r &\geq 0, \ \forall r \in \mathcal{R}_\Omega, \ \forall \Omega \in W & (12.56) \end{aligned}$$

Ejemplo 12.10 (formulación basada en flujos). Para ilustrar las restricciones anteriores, considérese el problema de una ciudad con una ronda de circunvalación y distintas rutas que pasan por el centro de dicha ciudad, que se representa gráficamente en la figura 12.9. Supóngase que se hacen 4000 viajes desde A hasta B, y 2500 viajes desde A hasta C. Las rutas posibles que

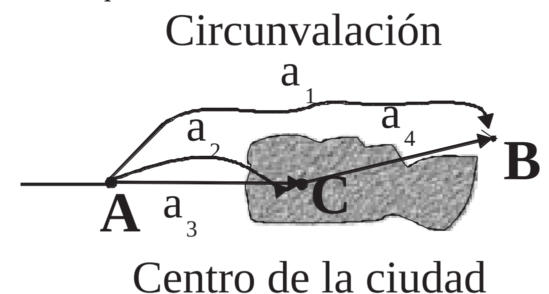


Figura 12.9: Diagrama que muestra las posibles rutas.

satisfacen el par $\Omega_1 = (A, B)$ de demanda son $r_1 = \{a_1\}$, $r_2 = \{a_2, a_4\}$, y $r_3 = \{a_3, a_4\}$, y las rutas para el par $\Omega_2 = (A, C)$ son $r_4 = \{a_2\}$ y $r_5 = \{a_3\}$. En este ejemplo, se tiene $W = \{1, 2\}$, y $\mathcal{R}_{\Omega_1} = \{r_1, r_2, r_3\}$ y $\mathcal{R}_{\Omega_2} = \{r_4, r_5\}$. Las variables de flujo en los caminos son $h_1, \ldots, h_5$, y las variables de flujo en los arcos son $f_1, \ldots, f_4$.

Las restricciones en este ejemplo son:

- Restricciones (12.51)

$$
\begin{aligned}
h_1 + h_2 + h_3 &= 4000 \\
h_4 + h_5 &= 2500
\end{aligned}
$$

- Restricciones (12.53)

$$
\begin{aligned}
h_1 &= f_1 \\
h_2 + h_4 &= f_2 \\
h_3 + h_5 &= f_3 \\
h_2 + h_3 &= f_4
\end{aligned}
$$

- Restricciones (12.52):

$$
h_1, \ldots, h_5 \geq 0
$$

■

Derivación de las condiciones de equilibrio del PAT

A continuación se demuestra que el modelo matemático anterior es adecuado para la consideración de las condiciones de equilibrio de Wardrop. La primera observación es que

$$
\begin{aligned}
C_a(f_a) &= \int_0^{f_a} c_a(x)dx \\
C'_a(f_a) &= c_a(f_a) \\
C''_a(f_a) &= c'_a(f_a)
\end{aligned}
$$

Como la función $c_a(f_a)$, que representa el coste del arco según su nivel de servicio, es una función no decreciente, se tiene que $c'_a(f_a) \geq 0$ y que la función $C_a(f_a)$ es una función convexa. Puesto que las restricciones son lineales, el problema es un problema de programación matemática convexo. Por tanto, las condiciones de KKT son necesarias y suficientes. Gracias a estas condiciones se demostrará que la solución óptima del PAT satisface las condiciones de equilibrio.

Al expresar en la función objetivo (12.54), los flujos en los arcos f_a en función de los flujos de los caminos (según la ecuación (12.55)), se puede formular el problema en función de las variables de flujo en los caminos, $\{h_r : r \in \mathcal{R}_\Omega,\ \Omega \in W\}$, como sigue. Minimizar

$$Z = \sum_{a \in \mathcal{A}} \int_0^{\sum\limits_{\Omega \in W} \sum\limits_{r \in \mathcal{R}_\Omega} \delta_{ar} h_r} c_a(x)dx$$

sujeto a

$$\begin{array}{rcll} \sum\limits_{r \in \mathcal{R}_\Omega} h_r & = & g_\Omega,\ \forall \Omega \in W & (12.57) \\ h_r & \geq & 0,\ \forall r \in \mathcal{R}_\Omega,\ \forall \Omega \in W & (12.58) \end{array}$$

La función lagrangiana de este problema es

$$\mathcal{L}(\mathbf{h}, \boldsymbol{\lambda}, \boldsymbol{\psi}) = Z + \sum_{\Omega \in W} \lambda_\Omega \left(g_\Omega - \sum_{r \in \mathcal{R}_\Omega} h_r \right) + \sum_{\Omega \in W} \sum_{r \in \mathcal{R}_\Omega} \psi_r(-h_r)$$

y las condiciones KKT (8.3) son

$$\frac{\partial \mathcal{L}}{\partial h_{r'}} = \sum_{a \in \mathcal{A}} c_a \left(\sum_{\Omega \in W} \sum_{r \in \mathcal{R}_\Omega} \delta_{ar} h_r \right) \delta_{ar'} - \lambda_{\Omega'} - \psi_{r'} = 0 \tag{12.59}$$

donde $r' \in \mathcal{R}_{\Omega'}$. Obsérvese que $\sum_{\Omega \in W} \sum_{r' \in \mathcal{R}_\Omega} \delta_{ar} h_{r'}$ es el flujo en un arco a, y $\sum_{a \in \mathcal{A}} c_a(f_a)\delta_{ar'}$ es la suma del coste asociado a todos los arcos de la ruta r', concretamente, ésta es el coste del camino r'. Si esta cantidad se denota como $C^*_{r'}$, la condición (12.59) se transforma en

$$C^*_{r'} = \lambda_{\Omega'} + \psi_{r'}$$

De la condición de holgura se tiene que $\psi_r h_r = 0$; por tanto, $h_r = 0$ ó $\psi_r = 0$. Si $h_{r'} > 0$, entonces $\psi_{r'} = 0$ y $C^*_{r'} = \lambda_{\Omega'}$. Si no, debido a la no negatividad del multiplicador $\psi_{r'}$, se tiene que $C^*_{r'} = \lambda_{\Omega'} + \psi_{r'} \geq \lambda_{\Omega'}$. Es decir, la condición anterior asegura que el coste de los caminos sin flujo es superior al coste de los caminos con flujo. Esta condición también asegura que en el óptimo, el coste de los caminos empleados por la demanda g_Ω tienen igual coste (y menor o igual que el coste del resto de los caminos del par Ω).

Implementación en GAMS

A continuación se implementa en GAMS la formulación PAT. El ejemplo es el mismo que se presentó en el apartado 12.6.2 e ilustró en la figura 12.8. Este ejemplo emplea las funciones BPR como costes de los arcos, obteniéndose la siguiente función objetivo

$$\begin{aligned} Z &= \sum_{a\in\mathcal{A}} C_a(f_a) = \sum_{a\in\mathcal{A}} \int_0^{f_a} c_a(x)dx = \sum_{a\in\mathcal{A}} \int_0^{f_a} \left[c_a^0 + b_a \left(\frac{x}{k_a}\right)^{n_a}\right] dx \\ &= \sum_{a\in\mathcal{A}} \left[c_a^0 f_a + \frac{b_a}{n_a+1}\left(\frac{f_a}{k_a}\right)^{n_a+1}\right] = \sum_{a\in\mathcal{A}} \left(c_a^0 f_a + d_a {f_a}^{m_a}\right) \end{aligned}$$

donde

$$d_a = \frac{b_a}{(n_a+1)k_a^{n_a+1}} \;\text{ y }\; m_a = n_a + 1$$

Los parámetros de la red ND se presentan en la tabla 12.9.

Tabla 12.9: Parámetros de las funciones de coste de la red Nguyen–Dupuis

Arco	c_a^0	d_a	m_a	Arco	c_a^0	d_a	m_a
(1,5)	7.0	0.00625	2.0	(1,12)	9.0	0.00500	2.0
(4,5)	9.0	0.00500	2.0	(4,9)	12.0	0.00250	2.0
(5,6)	3.0	0.00375	2.0	(5,9)	9.0	0.00375	2.0
(6,7)	5.0	0.00625	2.0	(6,10)	13.0	0.00250	2.0
(7,8)	5.0	0.00625	2.0	(7,11)	9.0	0.00625	2.0
(8,2)	9.0	0.00625	2.0	(9, 10)	10.0	0.00250	2.0
(9,13)	9.0	0.00250	2.0	(10,11)	6.0	0.00125	2.0
(11,2)	9.0	0.00250	2.0	(11,3)	8.0	0.00500	2.0
(12,6)	7.0	0.00125	2.0	(12,8)	14.0	0.00500	2.0
(13,3)	11.0	0.00500	2.0				

El fichero de entrada GAMS es el que sigue:

```
$title PROBLEMA DE ASIGNACION DE TRAFICO.

** En primer lugar se definen los conjuntos.
** El conjunto N representa los nudos de la red de trafico.
** El conjunto A(N,N) representa el conjunto de arcos.
** El conjunto W representa los pares O--D

SET
N conjunto de nudos /I1*I13/
W pares        /W1*W4/
A(N,N) conjunto de arcos
/I1.(I5,I12)
I4.(I5,I9)
```

```
I5.(I6,I9)
I6.(I7,I10)
I7.(I8,I11)
I8.(I2)
I9.(I10,I13)
I10.(I11)
I11.(I2,I3)
I12.(I6,I8)
I13.(I3)/

ODE(N,W) conjunto de origenes por demanda
/I1.(W1,W2)
I4.(W3,W4)/

DDE(N,W) conjunto de destinos por demanda
/I2.(W1,W3)
I3.(W2,W4)/;

** El conjunto de nudos N se duplica para referenciar distintos
elementos
** dentro de la misma restriccion.

ALIAS(N,I)
ALIAS(N,J)

PARAMETER
G(W) demanda en el par O--D de W
/W1 400
W2 800
W3 600
W4 200/;

TABLE  CDATA(N,N,*)  parametros de los costes de los arcos
          C0      D       M
  I1.I5    7.0  0.00625   2.0
  I1.I12   9.0  0.00500   2.0
  I4.I5    9.0  0.00500   2.0
  I4.I9   12.0  0.00250   2.0
  I5.I6    3.0  0.00375   2.0
  I5.I9    9.0  0.00375   2.0
  I6.I7    5.0  0.00625   2.0
  I6.I10  13.0  0.00250   2.0
  I7.I8    5.0  0.00625   2.0
  I7.I11   9.0  0.00625   2.0
  I8.I2    9.0  0.00625   2.0
  I9.I10  10.0  0.00250   2.0
  I9.I13   9.0  0.00250   2.0
  I10.I11  6.0  0.00125   2.0
  I11.I2   9.0  0.00250   2.0
  I11.I3   8.0  0.00500   2.0
  I12.I6   7.0  0.00125   2.0
  I12.I8  14.0  0.00500   2.0
  I13.I3  11.0  0.00500   2.0;

** Se declaran las variables de optimizacion.

VARIABLES
```

```
z         variable del valor de la funcion objetivo
f(I,J)    flujo en el arco I-J
fc(W,I,J) flujo del producto W en el arco I-J
cos(I,J)  coste del arco I-J en el equilibrio;

POSITIVE VARIABLE fc(W,N,N);

** Se declaran las restricciones.

EQUATIONS
COSTZ Funcion objetivo
BALANCE(W,I) ley de conservacion del flujo para el producto W.
FLOW(I,J) flujo total por los arcos
COST(I,J) coste del arco I-J;

** La funcion objetivo es la suma de los costes totales
** en todos los arcos.

COSTZ.. z=E= SUM((I,J)$A(I,J),CDATA(I,J,'CO')*f(I,J)
              +CDATA(I,J,'D')*f(I,J)**CDATA(I,J,'M'));
BALANCE(W,I) .. SUM(J$A(I,J),fc(W,I,J))-SUM(J$A(J,I),fc(W,J,I))
              =E=G(W)$ODE(I,W)-G(W)$DDE(I,W);
FLOW(I,J)$A(I,J)..SUM(W,fc(W,I,J))=E=f(I,J);
COST(I,J)$A(I,J)..cos(I,J)=E=CDATA(I,J,'CO')
             +2*CDATA(I,J,'D')*f(I,J);

** Los comandos siguientes definen el problema ND (Nguyen-Dupis),
** considerando todas las restricciones anteriores, y solicitan
** que GAMS resuelva el problema mediante un optimizador de pnl.

MODEL ND /ALL/;
SOLVE ND USING nlp MINIMIZING z;
```

El código anterior calcula los flujos y costes de los arcos en el equilibrio, que se muestran en la tabla 12.10. La información obtenida – el nivel del servicio de los arcos de transporte de la red – es esencial para la planificación futura del transporte en esa red.

Tabla 12.10: Flujos y costes de los arcos en el equilibrio

Arco	f_a	$c_a(f_a)$	Arco	f_a	$c_a(f_a)$
(1,5)	675.144	15.439	(1,12)	524.856	14.249
(4,5)	102.571	10.026	(4,9)	697.429	15.487
(5,6)	416.187	6.121	(5,9)	361.528	11.711
(6,7)	356.416	9.455	(6,10)	184.626	13.923
(7,8)	102.571	6.282	(7,11)	253.845	12.173
(8,2)	502.571	15.282	(9,10)	497.429	12.487
(9,13)	561.528	11.808	(10.I11)	682.056	7.705
(11.I2)	497.429	11.487	(11.I3)	438.472	12.385
(12.I6)	124.856	7.312	(12.I8)	400.000	18.000
(13.I3)	561.528	16.615			

Tabla 12.11: Caminos usados en el equilibrio para el PAT

Par O–D	Caminos empleados	h_r	C_r
$\Omega_1 = (1-2)$	$r_1 = 1-12-8-2$	400	47.53
$\Omega_2 = (1-3)$	$r_2 = 1-5-9-13-3$	361.528	55.57
	$r_3 = 1-5-6-7-11-3$	h_3	55.62
	$r_4 = 1-5-6-10-11-3$	h_4	55.57
	$r_5 = 1-12-6-7-11-3$	h_5	55.62
	$r_6 = 1-12-6-10-11-3$	h_6	55.56
$\Omega_3 = (4-2)$	$r_7 = 4-5-6-7-8-2$	102.571	47.17
	$r_8 = 4-9-10-11-2$	184.626	47.16
$\Omega_4 = (4-3)$	$r_9 = 4-9-13-3$	200	43.91

Se puede comprobar cómo se satisfacen las condiciones de equilibrio. A partir de los flujos obtenidos para cada producto, se pueden calcular los caminos elegidos y sus costes en el equilibrio. Esta información se muestra en la tabla 12.11. Para cada par O–D se muestra cuál es el trayecto óptimo junto con sus costes y flujos asociados.

Obsérvese que las rutas elegidas para el mismo producto tienen aproximadamente el mismo coste (véanse, por ejemplo, los productos Ω_2, Ω_3). Otra observación es que los flujos en los arcos, en el equilibrio, son únicos, y esto no ocurre con los flujos en los caminos. En este ejemplo, los flujos en las rutas que cumplen

$$\begin{aligned} h_5 + h_6 &= 124.856 \\ h_3 + h_4 &= 313.616 \\ h_5 + h_3 &= 253.845 \\ h_4 + h_6 &= 184.626 \end{aligned}$$

generan flujos en equilibrio en los arcos.

12.6.4 Modelos de asignación con restricciones laterales

A veces, en aplicaciones de redes de transporte, es necesario considerar *redes con limitaciones de capacidad*, en las que se presentan las siguientes restricciones:

$$s_k(\mathbf{f}) \leq 0, \quad \forall k \in \mathcal{K}$$

donde el conjunto $\mathcal{K}$ puede, por ejemplo, estar formado por arcos, nudos, rutas o cualquier combinación de éstos. Un ejemplo de esta clase de restricciones es

$$0 \leq f_a \leq u_a, \quad \forall a \in \mathcal{B} \subset \mathcal{A} \tag{12.60}$$

que forman parte del *problema de asignación de tráfico con capacidad limitada*, PATC.

Al diseñar una red futura, según el nivel del servicio que se necesite, los ingenieros deben determinar la capacidad de las carreteras y cruces en función del ancho de la carretera, del número de carriles, de las señalizaciones y de otros factores.

Si se emplea el PAT para evaluar el nivel del servicio, las predicciones que realiza este modelo de los flujos pueden ser, para algunos arcos, mayores que la capacidad que pueden soportar. Esto se debe a que el PAT permite que el volumen de tráfico en un arco sea infinito. Por ello, resulta obvio que restricciones como las (12.60) son fundamentales para la modelación del tráfico.

Para este nuevo problema se pueden formular condiciones de equilibrio similares al primer principio de Wardrop. Supóngase que las rutas elegidas para una demanda Ω están numeradas en orden creciente de costes, y que se denotan mediante c_i^Ω; $i = 1, 2, \ldots, l$. Asimismo, se supone que el número de rutas es l, y que las m primeras están *saturadas* de forma que al menos uno de sus arcos tiene un flujo igual a la capacidad máxima del arco. Existe equilibrio si y sólo si

$$c_1^\Omega \leq c_2^\Omega, \ldots \leq c_m^\Omega \leq c_{m+1}^\Omega = \cdots c_l^\Omega, \quad \forall \Omega \in W$$

A continuación, se plantean dos alternativas para tratar estas restricciones adicionales:

1. Considerar explícitamente el límite máximo de los flujos en los arcos (12.60). Esta alternativa equivale a modificar los costes de los arcos de la siguiente forma

$$\hat{c}_a(f_a) = c_a(f_a) + \phi_a(f_a), \quad \forall a \in \mathcal{B}$$

donde las funciones

$$\phi_a(f_a) = \begin{cases} +\infty & \text{si } f_a > k_a \\ 0 & \text{si } f_a \leq k_a \end{cases}$$

no son continuas, por lo que el problema se complica.

2. Emplear una función de coste, que tiende a infinito si los flujos en los arcos se aproximan al límite máximo. Esta alternativa se asemeja a los métodos barrera y consiste en aproximar la función ϕ_a mediante una función continua y diferenciable. Por ejemplo, una aproximación válida es

$$\hat{c}_a(f_a) = c_a(f_a) + s_a \ln\left(1 + \frac{q_a}{k_a - f_a}\right), \forall a \in \mathcal{B}$$

donde $s_a \to 0$, y k_a es la capacidad máxima del arco a. Esta función no está definida para aquellos arcos en los que su flujo supera la capacidad del mismo, por lo que se necesita un procedimiento posterior que recupere las infactibilidades que se originen. Este hecho puede provocar errores en la resolución del problema. Para evitar estas dificultades, las funciones ϕ_a

se aproximarán mediante la fórmula BPR, $s_a\left(f_a/k_a\right)^{n_a}$, obteniéndose las siguientes funciones de coste de los arcos

$$\bar{c}_a(f_a) = c_a(f_a) + s_a \left(\frac{f_a}{k_a}\right)^{n_a}, \quad \forall a \in \mathcal{B} \tag{12.61}$$

donde $n_a \to +\infty$.

Ejemplo 12.11 (red restringida). Para ilustrar las dos alternativas, se presenta un ejemplo en el que se incluyen las restricciones adicionales en la red ND

$$\begin{aligned} f_{10,11} &\leq 400 \\ f_{12,8} &\leq 300 \end{aligned}$$

Este ejemplo se denomina ND-C, y el conjunto $\mathcal{B}$ es $\{(10,11),(12,8)\}$. En el enfoque implícito se utilizan los valores $n_a = 9$, y $s_a = 20$, $\forall a \in \mathcal{B}$.

A continuación, se presentan sólo los comandos de GAMS adicionales que se necesitan para resolver el problema PATC, partiendo del código del PAT.

```
$title PROBLEMA DE ASIGNACION DE TRAFICO RESTRINGIDO.

** El conjunto B(N,N) es un nuevo conjunto con los arcos restringidos

SET
...

B(N,N) conjunto con los arcos restringidos
                /I10.I11
                I12.I8/;
PARAMETER
U(I,J) limite superior del arco I--J
                /I10.I11 400
                 I12.I8  300/
...

** Se establece el limite superior de las variables de flujo

VARIABLES
...
f.up(I,J)$B(I,J)=U(I,J);
```

En la tabla 12.12 se presentan los flujos en los arcos, y los costes en el equilibrio para los dos enfoques considerados. Empleando los flujos en los arcos de cada producto, se han calculado las rutas elegidas y sus costes. Esta información también se muestra en la tabla 12.12. Se puede comprobar cómo se satisfacen las condiciones de equilibrio. Los pares Ω_1 y Ω_3 ilustran el hecho de que las rutas saturadas tienen asociado un coste menor que las no saturadas. Es más, el hecho de que rutas no saturadas tengan el mismo coste se cumple para los pares Ω_1 y Ω_2. En la tabla 12.13 se ha calculado el coste de la ruta mediante $c_a(f_a)$ y no mediante $\bar{c}_a(f_a)$. Es importante destacar que el coste de los arcos

restringidos es excesivamente grande para el enfoque implícito (véase la tabla 12.12). En consecuencia, en el enfoque implícito no se cumplen las condiciones de equilibrio con la aproximación obtenida. Esto se debe a que los costes de los arcos restringidos son altamente no lineales, lo que representa una desventaja con respecto al enfoque implícito.

Desde la perspectiva de modelado, el enfoque explícito es ventajoso al permitir que los flujos en los arcos puedan alcanzar límites (pero no sobrepasarlos), mientras que si se emplean funciones de tiempo de viaje, como (12.61), conduce (para este ejemplo) a que todos los flujos en los arcos sean estrictamente mayores que los límites (véase la tabla 12.12). Obsérvese que ambos enfoques generan las mismas rutas, pero se diferencian en cuanto a la demanda y el coste. Se puede concluir diciendo que las predicciones obtenidas por ambos enfoques no son significativamente distintas.

Una ventaja relevante del modelado explícito es que permite interpretar los multiplicadores de Lagrange asociados a las restricciones de capacidad máxima. En la tabla 12.12 se muestran los multiplicadores de Lagrange para este ejemplo. Estos parámetros miden el tiempo que ganan los viajeros por las rutas saturadas respecto al que emplearían por otras rutas disponibles que fuesen más rápidas. Por ejemplo, la ruta r_1 está saturada porque contiene el arco $(12, 8)$. El multiplicador de Lagrange correspondiente a este arco es 11.501, que es aproximadamente igual a la diferencia entre los costes de las rutas saturadas y las no saturadas, es decir, $58.70 - 47.20$.

Esta interpretación demuestra que se pueden determinar los flujos en equilibrio con arcos con capacidad limitada si se resuelve el problema no restringido (PAT) correspondiente, en el que las funciones de coste son $\hat{c}_a(f_a) = c_a(f_a) + \beta_a$ para todo $a \in \mathcal{B}$, donde β_a son los multiplicadores de Lagrange. El lector puede comprobar que si en este ejemplo se usan los costes de los arcos $\hat{c}_a$, las rutas elegidas satisfacen el primer principio de Wardrop.

Como conclusión, el enfoque implícito presenta la ventaja computacional de que emplea un modelo de asignación sin restricciones de capacidad, pero tiene el inconveniente de sobrecargar las rutas restringidas, y dificulta la interpretación de los multiplicadores de Lagrange. ∎

12.6.5 El caso de la demanda elástica

El PAT se ha formulado considerando que la demanda es fija, pero es más realista considerar la naturaleza elástica de la demanda. Los viajeros tienen un número limitado de posibles rutas y se guían por motivos económicos cuando toman sus decisiones. Por ejemplo, cuando aumenta la congestión, algunos usuarios de vehículos privados prefieren desplazarse mediante otros modos de transporte (por ejemplo, el metro).

Para tener en cuenta la elasticidad de la demanda, se puede suponer que el número de viajes, g_Ω, para el par Ω de la red es una función de su coste generalizado de viaje:

$$g_\Omega = G_\Omega(c_\Omega)$$

Tabla 12.12: Flujos y costes de los arcos en el equilibrio para el PATC, y los multiplicadores de Lagrange (β_a) correspondientes.

	Enfoque explícito		Enfoque implícito	
Arco	f_a	$c_a(f_a)$	f_a	$c_a(f_a)$
(1,5)	689.920	15.624	690.242	15.628
(1,12)	510.080	14.101	509.758	14.098
(4,5)	200.000	11.000	178.735	10.787
(4,9)	600.000	15.000	621.265	15.106
(5,6)	400.266	6.002	393.659	5.952
(5,9)	489.655	12.672	475.318	12.565
(6,7)	610.345	12.629	586.810	12.335
(6,10)	.	13.000	.	13.000
(7,8)	267.749	8.347	253.988	8.175
(7,11)	342.597	13.282	332.822	13.160
(8,2)	567.749	16.097	570.595	16.132
(9,10)	400.000	12.000	421.265	12.106
(9,13)	689.655	12.448	675.318	12.377
(10,11)	400.000	7.000	421.265	7.053
	$\beta_a = 8.914$			$\bar{c}_a = 342.775$
(11,2)	432.251	11.161	429.405	11.147
(11,3)	310.345	11.103	324.682	11.247
(12,6)	210.080	7.525	193.151	7.483
(12,8)	300.000	17.000	316.607	17.166
	$\beta_a = 11.501$			$\bar{c}_a = 359.950$
(13,3)	689.655	17.897	675.318	17.753

Tabla 12.13: Flujo y coste de las rutas en el equilibrio para el PATC

		Enfoque			
		Explícito		Implícito	
Par O–D	Rutas elegidas	h_r	C_r	h_r	C_r
Ω_1	$r_1 = 1 - 12 - 8 - 2$	300	47.20	316.6	47.40
(1-2)	$r_2 = 1 - 12 - 6 - 7 - 8 - 2$	67.75	58.70	75.25	58.23
	$r_3 = 1 - 12 - 6 - 7 - 11 - 2$	32.25	58.70	8.14	58.23
Ω_2	$r_4 = 1 - 5 - 6 - 7 - 11 - 3$	200.3	58.64	214.9	58.33
(1-3)	$r_5 = 1 - 5 - 9 - 13 - 3$	489.7	58.64	475.3	58.33
	$r_6 = 1 - 12 - 6 - 7 - 11 - 3$	110.1	58.64	109.8	58.32
Ω_3	$r_9 = 4 - 9 - 10 - 11 - 2$	400	45.10	421.3	45.42
(4-2)	$r_{10} = 4 - 5 - 6 - 7 - 8 - 2$	200	54.07	178.74	53.39
Ω_4	$r_{11} = 4 - 9 - 13 - 3$	200	45.34	200	45.24
(4-3)					

donde c_Ω es el mínimo coste del viaje para el par Ω, y G_Ω es la función de demanda. También se supone que G_Ω es una función no negativa, continua, y estrictamente decreciente para cada $\Omega \in W$. La inversa de esta función proporciona el número de viajes en términos del coste del viaje, es decir, $c_\Omega = G_\Omega^{-1}(g_\Omega)$.

La condición de equilibrio asegura que el viaje entre los pares O–D satisface la función de demanda, y que el tiempo de viaje empleado entre cualquier par O–D es el mismo, y es menor o igual que el tiempo de viaje empleado en cualquier ruta no elegida.

El modelo siguiente incorpora al problema de asignación una demanda elástica. Las condiciones de equilibrio anteriores se obtienen al resolver el siguiente problema (PATE). Minimizar

$$Z = \sum_{(i,j)\in\mathcal{A}} \int_0^{f_{ij}} c_{ij}(x)dx - \sum_{\Omega\in W} \int_0^{g_\Omega} G_\Omega^{-1}(x)dx$$

sujeto a

$$\begin{aligned} \sum_{j\in A(i)} f_{ij}^\Omega - \sum_{j\in B(i)} f_{ji}^\Omega &= r_i^\Omega,\ \forall i \in \mathcal{N},\quad \forall\Omega \in W && (12.62) \\ \sum_{\Omega\in W} f_{ij}^\Omega &= f_{ij}, \forall(i,j) \in \mathcal{A} && (12.63) \\ f_{ij}^\Omega &\geq 0 \quad \forall\Omega \in W, \forall(i,j) \in \mathcal{A} && (12.64) \end{aligned}$$

donde

$$r_i^\Omega = \begin{cases} g_\Omega & \text{si } O^\Omega = i \\ -g_\Omega & \text{si } D^\Omega = i \\ 0 & \text{si } i \text{ es un nudo intermedio para el producto } \Omega \end{cases}$$

Obsérvese que en el PATE los términos r_i^Ω son variables, mientras que en el PAT son constantes.

Para ilustrar el PATE se presenta el siguiente caso de estudio. Se supone que todos los viajeros pueden elegir entre distintos medios de transporte, y que se tiene una función de demanda (modelo "logit") que proporciona el número de viajes que se realizan con cada medio, g_Ω^k, cuya expresión es

$$g_\Omega^k = G_\Omega^k(\mathbf{c}_\Omega) = \frac{\exp[-\left(\alpha^k + \beta_1 c_\Omega^k\right)]}{\sum\limits_{k'} \exp[-\left(\alpha^{k'} + \beta_1 c_\Omega^{k'}\right)]} \bar{g}_\Omega \qquad (12.65)$$

donde c_Ω^k es la percepción que tiene el viajero de los costes generalizados de viajar entre los elementos del par $\Omega = (i,j)$ usando el medio k, que se corresponde con la elección de la ruta óptima en la red; $\{\mathbf{c}_\Omega\}$ es el vector de costes generalizados para todos los medios de transporte disponibles, $\bar{g}_\Omega$ es la demanda total (en cualquier medio de transporte) para el par Ω (O–D) para cualquier medio;

y α^k y β_1 son los parámetros del modelo "logit". Para dos posibles medios, como el coche (a) y el transporte público (b) la función (12.65) se simplifica

$$G^a_\Omega(\mathbf{c}_\Omega) = \frac{1}{1+\exp[-\left(\alpha^{ab}+\beta_1(c^b_\Omega - c^a_\Omega)\right)]}\bar{g}_\Omega$$

donde $\alpha^{ab} = \alpha^b - \alpha^a$. Se supone que el coste del viaje c^b_Ω usando el transporte público es independiente del volumen de tráfico, y, por tanto, es constante. La función inversa del modelo "logit" es

$$c^a_\Omega = G^{-1}(g^a_\Omega) = c^b_\Omega + \frac{1}{\beta_1}\left[\alpha^{ab} + \log(\bar{g}_\Omega - g^a_\Omega) - \log(g^a_\Omega)\right]$$

Mediante la relación $g^a_\Omega + g^b_\Omega = \bar{g}_\Omega$, se obtiene la siguiente igualdad

$$-\int_0^{g^a_\Omega} G^{-1}(x)dx = c^b_\Omega g^b_\Omega + \left(\frac{1}{\beta_1}\right)\sum_{k\in\{a,b\}} g^k_\Omega(\log g^k_\Omega - 1 + \alpha^k) + C$$

donde C es una constante. Por tanto, la función objetivo es

$$\sum_{(i,j)\in\mathcal{A}}\int_0^{f_{ij}} c_{ij}(x)dx + \sum_{\Omega\in W} c^b_\Omega g^b_\Omega + \left(\frac{1}{\beta_1}\right)\sum_{\Omega\in W}\sum_{k\in\{a,b\}} g^k_\Omega(\log g^k_\Omega - 1 + \alpha^k)$$

Ejemplo 12.12 (caso de demanda elástica). Para ilustrar el modelo PATE se considera una red de metro. Supóngase que existen suficientes conexiones como para satisfacer los cuatro pares de demanda y que el tiempo de viaje es independiente del flujo (véase la tabla 12.14).

Tabla 12.14: Datos de entrada del PATE

par O–D	$\bar{g}_\Omega$	c^b_Ω	Parámetros "logit"
Ω_1	600	41	$\alpha^a = -2.0$
Ω_2	1000	46	$\alpha^b = 0.0$
Ω_3	800	43	$\beta_1 = 0.1$
Ω_4	400	40	

El siguiente código es la implementación de este modelo en GAMS.

```
$title PROBLEMA DE ASIGNACION DE TRAFICO CON DEMANDA ELASTICA.

SET
N conjunto de nudos /I1*I13/
W pares         /W1*W4/
A(N,N) conjunto de ramas
/I1.(I5,I12)
I4.(I5,I9)
I5.(I6,I9)
I6.(I7,I10)
I7.(I8,I11)
I8.(I2)
```

```
I9.(I10,I13)
I10.(I11)
I11.(I2,I3)
I12.(I6,I8)
I13.(I3)/

ODE(N,W) conjunto de origenes por demanda
/I1.(W1,W2)
I4.(W3,W4/

DDE(N,W) conjunto de destinos por demanda/I2.(W1,W3)
I3.(W2,W4)/;

ALIAS(N,I)
ALIAS(N,J)

PARAMETER
G(W) demanda en el par O--D de W
/W1 600
W2 1000
W3 800
W4 400/

Cb(W) coste del viaje en transporte publico para el par O--D de W
/W1 41
W2 35
W3 43
W4 40/

BETA
ALPHAa
ALPHAb;
BETA=0.1;
ALPHAa=-2.;
ALPHAb=0.;

TABLE  CDATA(N,N,*) parametros del coste de los arcos
           C0      D
  I1.I5    7.0  0.00625
  I1.I12   9.0  0.005
  I4.I5    9.0  0.005
  I4.I9   12.0  0.0025
  I5.I6    3.0  0.00375
  I5.I9    9.0  0.00375
  I6.I7    5.0  0.00625
  I6.I10  13.0  0.0025
  I7.I8    5.0  0.00625
  I7.I11   9.0  0.00625
  I8.I2    9.0  0.00625
  I9.I10  10.0  0.0025
  I9.I13   9.0  0.0025
  I10.I11  6.0  0.00125
  I11.I2   9.0  0.0025
  I11.I3   8.0  0.005
  I12.I6   7.0  0.00125
  I12.I8  14.0  0.005
  I13.I3  11.0  0.005;
```

```
** Se declaran las variables de optimizacion.

VARIABLES
  z          variable de la funcion objetivo
  f(N,N)     flujo en los arcos
  fc(W,N,N)  flujo en los arcos del producto W
  cos(N,N)   coste del arco en el equilibrio
  ga(W)      demanda de coche
  gb(W)      demanda de transporte publico;

POSITIVE VARIABLE fc(W,N,N);

ga.LO(W)=0.01;
gb.LO(W)=0.01;

** Se declaran las restricciones.

EQUATIONS
  COST Funcion objetivo
  BALANCE(W,I) condicion de conservacion del flujo para el producto W.
  FLOW(I,J) flujo total en los arcos
  MODAL(W)  Particion modal de la demanda
  COSTE(I,J) coste en el equilibrio;

COST .. z=E= SUM((I,J)$A(I,J),CDATA(I,J,'CO')*f(I,J)
              +CDATA(I,J,'D')*f(I,J)**CDATA(I,J,'M'))
+SUM(W,Cb(W)*gb(W))+1/BETA* SUM(W,ga(W)*(-1+ALPHAa+LOG(ga(W)))
              + gb(W)*(-1+ALPHAb+LOG(gb(W))) );
BALANCE(W,I) .. SUM(J$A(I,J),fc(W,I,J))-SUM(J$A(J,I),fc(W,J,I))
                 =E=ga(W)$ODE(I,W)-ga(W)$DDE(I,W);
FLOW(I,J)$A(I,J)..SUM(W,fc(W,I,J))=E=f(I,J);
COSTE(I,J)$A(I,J)..cos(I,J)=E=CDATA(I,J,'CO')
                              +2*CDATA(I,J,'D')*f(I,J);
MODAL(W).. G(W)=E=ga(W)+gb(W);
MODEL nd /ALL/;
SOLVE nd USING nlp MINIMIZING z;
```

Los resultados obtenidos se muestran en la tabla 12.15.

Tabla 12.15: Coste en el equilibrio de PATE y división modal.

Par O–D	c_Ω^a	c_Ω^b	g_Ω^a	g_Ω^b
Ω_1	49.29	41	459.821	140.179
Ω_2	56.73	46	736.288	263.712
Ω_3	48.52	43	647.730	152.270
Ω_4	45.56	40	323.124	76.876

Es necesario realizar algunos comentarios sobre los resultados. Por ejemplo, supóngase que el operador del sistema de transporte incrementa la frecuencia de las líneas de transporte público para satisfacer el par de demanda Ω_2, y que con ello se reduce el tiempo de viaje a 35. Entonces, el operador calculará la

nueva demanda y la congestión que se origina en la red debido a la misma. Estos resultados se muestran en la tabla 12.16. Obsérvese que la misma del transporte público se incrementa para el par Ω_2. Esto implica una reducción en el nivel de congestión de la red de carreteras y la alternativa de transporte en coche resulta algo más atractiva para los demás pares.

Tabla 12.16: Coste en el equilibrio del PATE y partición modal con intervención sobre el sistema de transporte.

Par O–D	c^a_Ω	c^b_Ω	g^a_Ω	g^b_Ω
Ω_1	48.53	41	466.009	133.991
Ω_2	52.79	35	554.843	445.157
Ω_3	48.34	43	649.947	150.053
Ω_4	44.56	40	329.594	70.406

■

12.6.6 Combinación de asignación y distribución

Una manera adecuada de realizar el proceso de la planificación (explicado en el apartado 12.6.1) es combinar el mayor número de pasos en uno solo, en concreto, se combinan los pasos de asignación y distribución del esquema de cuatro etapas. Supóngase conocido el número de viajes que parten de los orígenes O_i, y el número de viajes que llega a los destinos D_j, pero que no se conoce la matriz de viajes O–D. Esta matriz, g_Ω, $\Omega = (i,j) \in W$, debe cumplir que

$$\begin{array}{rcl} \sum_j g_{ij} & = & O_i, \ \ \forall i \\ \sum_i g_{ij} & = & D_j, \ \ \forall j \end{array} \tag{12.66}$$

Los modelos de distribución se tienen en cuenta a la hora de predecir patrones de viaje, cuando se producen cambios importantes en la red considerada. Estos modelos consideran el comportamiento del viajero y cómo influyen factores externos, como el número de viajes realizados o la distancia recorrida, en su comportamiento. En estos modelos se supone que el número de viajes depende del coste del viaje y del potencial de cada zona en la que se inicia el mismo. Se suelen emplear expresiones del tipo

$$g_{ij} = p(c_{ij}) = \alpha O_i D_j f(c_{ij})$$

donde α es un factor de proporcionalidad, y $f(c_{ij})$ consta de uno o varios parámetros ajustables. Esta función se denomina *función de disuasión*. Las expresiones que habitualmente se usan son

$$\begin{array}{ll} p(c_{ij}) = \exp(-\beta c_{ij}) & \text{función exponencial} \\ p(c_{ij}) = c_{ij}^{-n} & \text{función potencial} \\ p(c_{ij}) = c_{ij}^{n} \exp(-\beta c_{ij}) & \text{función combinada} \end{array} \tag{12.67}$$

Supóngase que la función $p(c_{ij})$ es una función decreciente del coste del viaje, entonces el modelo anterior se puede formular mediante el siguiente problema de programación matemática (PATD). Minimizar

$$Z = \sum_{(i,j)\in\mathcal{A}} \int_0^{f_{ij}} c_{ij}(x)dx - \sum_{\Omega\in W} \int_0^{g_\Omega} p^{-1}(x)dx$$

sujeto a

$$\sum_j g_{ij} = O_i, \ \forall i \quad (12.68)$$

$$\sum_i g_{ij} = D_j, \ \forall j \quad (12.69)$$

$$\sum_{j\in A(i)} f_{ij}^{\Omega} - \sum_{j\in B(i)} f_{ji}^{\Omega} = r_i^{\Omega}, \ \forall i \in \mathcal{N}, \ \ \forall \Omega \in W \quad (12.70)$$

$$\sum_{\Omega\in W} f_{ij}^{\Omega} = f_{ij}, \ \forall (i,j) \in \mathcal{A} \quad (12.71)$$

$$f_{ij}^{\Omega} \geq 0 \ \ \forall \Omega \in W, \ \forall (i,j) \in \mathcal{A} \quad (12.72)$$

donde

$$r_i^{\Omega} = \begin{cases} g_\Omega & \text{si } O^\Omega = i \\ -g_\Omega & \text{si } D^\Omega = i \\ 0 & \text{si } i \text{ es un nudo intermedio para el producto } \Omega \end{cases}$$

El ejemplo siguiente ilustra el modelo PATD.

Ejemplo 12.13 (modelo de distribución gravitacional). Este ejemplo ilustra el problema PATD con un *modelo de distribución gravitacional*, que se obtiene usando la expresión exponencial de la función de disuasión. En este caso, se tiene que

$$\begin{aligned} -\int_0^{g_\Omega} p^{-1}(x)dx &= -\int_0^{g_\Omega} \left[-\frac{1}{\beta}\log(x) + (d_i + d_j + \alpha')\right] dx \\ &= \frac{1}{\beta} g_\Omega (\log g_\Omega - 1) + (d_i + d_j + \alpha') g_\Omega \end{aligned}$$

donde $d_i = \log O_i/\beta$, $d_j = \log D_j/\beta$, y $\alpha' = \alpha/\beta$. La función objetivo se expresa como

$$\sum_{(i,j)\in\mathcal{A}} \int_0^{f_{ij}} c_{ij}(\mathbf{x})d\mathbf{x} + \frac{1}{\beta}\sum_{\Omega\in W} g_\Omega (\log g_\Omega - 1) + \sum_{\Omega\in W} (d_i + d_j + \alpha') g_\Omega$$

El término $\sum_{\Omega\in W}(d_i + d_j + \alpha')g_\Omega$ es constante para el conjunto de soluciones factibles definido en (12.66) y, por tanto, se puede eliminar. Asimismo, se puede

demostrar que el problema de programación matemática planteado es convexo, y aplicando las condiciones de KKT se tiene que la solución óptima cumple

$$g_{ij}^{*} = A_i O_i B_j D_j \exp(-\beta c_{ij}^{*})$$

donde c_{ij}^{*} es el coste en el equilibrio. El conjunto de parámetros A_i y B_j sustituyen al factor de proporcionalidad α para que puedan cumplir las restricciones (12.66).

El siguiente fichero de entrada GAMS implementa el modelo anterior tomando los datos que se muestran en la tabla 12.17. Los resultados obtenidos se presentan en la tabla 12.18.

```
$title PROBLEMA DE ASIGNACION Y DISTRIBUCION DEL TRAFICO.

SET
  N conjunto de nudos /I1*I13/
  W pares         /W1*W4/
  O origenes      /I1,I4/
  D destinos      /I2,I3/

OW(O,W)
/I1.(W1,W2)
I4.(W3,W4)/
DW(D,W)
/I2.(W1,W3)
I3.(W2,W4)/

A(N,N) conjunto de arcos
/I1.(I5,I12)
I4.(I5,I9)
I5.(I6,I9)
I6.(I7,I10)
I7.(I8,I11)
I8.(I2)
I9.(I10,I13)
I10.(I11)
I11.(I2,I3)
I12.(I6,I8)
I13.(I3)/

ODE(N,W) conjunto de origenes por demanda
/I1.W1
I1.W2
I4.W3
I4.W4/
DDE(N,W) conjunto de destinos por demanda
/I2.W1
I3.W2
I2.W3
I3.W4/;

ALIAS(N,I)
ALIAS(N,J)

PARAMETER
Oi(O) numero de viajes que parten de Oi
```

```
/I1 1200
I4 800/

Dj(D) numero de viajes que llegan a Dj
/I2 1000
I3 1000/

BETA;
BETA=0.2;

TABLE  CDATA(N,N,*) parametros del coste de los arcos
          C0      D       M
  I1.I5    7.0  0.00625   2.0
  I1.I12   9.0  0.005     2.0
  I4.I5    9.0  0.005     2.0
  I4.I9   12.0  0.0025    2.0
  I5.I6    3.0  0.00375   2.0
  I5.I9    9.0  0.00375   2.0
  I6.I7    5.0  0.00625   2.0
  I6.I10  13.0  0.0025    2.0
  I7.I8    5.0  0.00625   2.0
  I7.I11   9.0  0.00625   2.0
  I8.I2    9.0  0.00625   2.0
  I9.I10  10.0  0.0025    2.0
  I9.I13   9.0  0.0025    2.0
  I10.I11  6.0  0.00125   2.0
  I11.I2   9.0  0.0025    2.0
  I11.I3   8.0  0.005     2.0
  I12.I6   7.0  0.00125   2.0
  I12.I8  14.0  0.005     2.0
  I13.I3  11.0  0.005     2.0;

VARIABLES
  z          variables de la funcion objetivo
  f(N,N)     flujo en los arcos
  fc(W,N,N)  es el flujo en los arcos debido a el producto W
  cos(N,N)   coste del arco en el equilibrio
  g(W)       demande en el par W;

POSITIVE VARIABLE fc(W,N,N);

g.LO(W)=0.01;

EQUATIONS
  COST  Funcion objetivo
  BALANCE(W,I) condicion de la conservacion del flujo para el producto
W.
  FLOW(I,J) flujo total en los arcos
  ORIGIN(O)
  DESTIN(D)
  COSTE(I,J) coste en el equilibrio;

COST .. z=E= SUM((I,J)$A(I,J),CDATA(I,J,'C0')*f(I,J)
              +CDATA(I,J,'D')*f(I,J)**CDATA(I,J,'M'))
              +(1/BETA)*SUM(W,g(W)*(LOG(g(W))-1 ));
BALANCE(W,I) .. SUM(J$A(I,J),fc(W,I,J))-SUM(J$A(J,I),fc(W,J,I))
                 =E=g(W)$ODE(I,W)-g(W)$DDE(I,W);
```

```
FLOW(I,J)$A(I,J)..SUM(W,fc(W,I,J))=E=f(I,J);
COSTE(I,J)$A(I,J)..cos(I,J)=E=CDATA(I,J,'CO')
                                +2*CDATA(I,J,'D')*f(I,J);
ORIGIN(O)..Oi(O)=E= SUM(W,g(W)$OW(O,W));
DESTIN(D)..Dj(D)=E= SUM(W,g(W)$DW(D,W));
MODEL nd /ALL/;
SOLVE nd USING nlp MINIMIZING z;
```

Tabla 12.17: Datos de entrada del modelo TAPD

Origen	O_i	Destino	D_j	β
1	1200	2	1000	0.2
4	800	3	1000	–

Tabla 12.18: Resultados para el modelo TAPD

par O–D	g_Ω	Coste en el equilibrio
Ω_1	650.312	52.21
Ω_2	549.688	53.65
Ω_3	349.688	46.77
Ω_4	450.312	45.05

■

12.7 Coordinación hidrotérmica a corto plazo

La coordinación hidrotérmica a corto plazo (CHTCP) determina la secuencia de arranques y paradas de las centrales térmicas, así como la potencia que deben producir las centrales hidráulicas y térmicas para satisfacer la demanda de los clientes con un nivel adecuado de seguridad, de forma que el coste total de operación sea mínimo. Debido a lo costoso que resulta arrancar las centrales térmicas, la decisión de qué centrales se deben emplear para cubrir la demanda al menor coste es compleja. Si el sistema de energía eléctrica considerado no incluye centrales hidráulicas, el problema anterior se denomina programación horaria de centrales térmicas.

Matemáticamente, el problema de la CHTCP se puede formular como un problema de optimización no lineal y entera-mixta. Para sistemas de energía eléctrica realistas este problema es también de gran dimensión. La resolución de este problema no lineal, de gran dimensión y combinatorio, no es una tarea fácil. Las técnicas de relajación lagrangiana (RL) resultan adecuadas a la hora de resolver este tipo de problemas (véase Muckstadt y Koening [77], Merlin y Sandrin [74], Bertsekas et al. [12], Zhuang y Galiana [107], Yan et al. [106],

Mendes et al. [73], Rakic y Marcovic [91], Wang et al. [101], Pellegrino et al. [85], Luh et al.[69] y Jiménez y Conejo [60], por ejemplo). Las técnicas de programación dinámica requieren una discretización de las variables continuas e hipótesis que simplifican drásticamente el problema para hacerlo computacionalmente tratable (Hobbs et al. [52]). Las técnicas de programación lineal entera-mixta no sólo linealizan el problema, sino que además lo simplifican notablemente a causa de su gran dimensión (véase Dillon et al. [32], Brannlund et al. [16], Medina et al. [70]).

Al utilizar las técnicas de RL para resolver el problema CHTCP, el problema primal relajado que resulta, se puede descomponer de manera natural en un subproblema por cada central térmica y en un subproblema por cada cuenca hidráulica. Por tanto, mediante las técnicas de RL, se obtiene la solución del problema CHTCP (problema de optimización complejo y de gran dimensión) mediante la resolución de varios problemas de pequeña dimensión y estructura similar.

La descomposición en subproblemas permite modelar de manera precisa cada generador así como seleccionar la técnica de optimización más adecuada a la estructura de cada subproblema. También permite aplicar técnicas de procesamiento paralelo, reduciéndose así el tiempo de cálculo.

Además de todas estas ventajas, que se derivan de la descomposición en subproblemas del problema primal relajado, la aplicación de las técnicas de RL para resolver el problema CHTCP conlleva una ventaja adicional: las variables del problema dual (los multiplicadores de Lagrange) tienen un significado económico muy útil en el ámbito de los mercados competitivos de energía eléctrica descentralizados, así como en sistemas centralizados.

12.7.1 Formulación del problema mediante RL

El problema CHTCP se puede formular como un problema de optimización no lineal combinatorio en el que los costes de operación se minimizan sujetos a que se cumplan las restricciones técnicas de las centrales térmicas e hidráulicas, y las restricciones de carga. Las restricciones de carga engloban a las restricciones de demanda de energía eléctrica por parte de los clientes y a las restricciones de reserva, que aseguran un nivel de seguridad adecuado.

Los principales elementos de este problema son:

1. **Datos.**

   I: número de centrales térmicas

   J: número de cuencas hidráulicas

   $\mathbf{H}$: vector de demandas

   $\mathbf{h}_i(\mathbf{x}_i)$: contribución de la central térmica i a satisfacer la demanda

   $\mathbf{h}_j(\mathbf{x}_j)$: contribución de la cuenca hidrográfica j a satisfacer la demanda

   $\mathbf{G}$: vector de reserva de potencia

   $\mathbf{g}_i(\mathbf{x}_i)$: contribución de la central térmica i a satisfacer la reserva de potencia

$\mathbf{g}_j(\mathbf{x}_j)$: contribución de la central hidráulica j a satisfacer la reserva de potencia

$\mathbf{H}$, $\mathbf{h}_i(\mathbf{x}_i)$, $\mathbf{h}_j(\mathbf{x}_j)$, $\mathbf{G}$, $\mathbf{g}_i(\mathbf{x}_i)$ y $\mathbf{g}_j(\mathbf{x}_j)$ son vectores cuya dimensión es igual al número de períodos de tiempo del horizonte temporal considerado.

2. **Variables.**

   $\mathbf{x}_i$: vector de variables asociado a la central térmica i

   $\mathbf{x}_j$: vector de variables asociado a la cuenca hidráulica j

3. **Restricciones.**

$$\begin{array}{rcl} \mathbf{s}_i(\mathbf{x}_i) & \leq & \mathbf{0}, i = 1, \ldots, I \\ \mathbf{s}_j(\mathbf{x}_j) & \leq & \mathbf{0}, j = 1, \ldots, J \\ \sum\limits_{i=1}^{I} \mathbf{h}_i(\mathbf{x}_i) + \sum\limits_{j=1}^{J} \mathbf{h}_j(\mathbf{x}_j) & = & \mathbf{H} \\ \sum\limits_{i=1}^{I} \mathbf{g}_i(\mathbf{x}_i) + \sum\limits_{j=1}^{J} \mathbf{g}_j(\mathbf{x}_j) & \leq & \mathbf{G} \end{array} \tag{12.73}$$

   El primer grupo de restricciones son las correspondientes a las centrales térmicas, el segundo, representa las correspondientes a las centrales hidráulicas, el tercero hace que se satisfagan las restricciones de demanda de los clientes, y el cuarto, representa las restricciones de reserva de potencia. En la formulación anterior queda implícito el tiempo.

4. **Función a optimizar.** La función objetivo representa los costes totales de operación (el coste de producción hidráulica es despreciable comparado con el coste de la producción térmica)

$$f(\mathbf{x}) = \sum_{i=1}^{I} f_i(\mathbf{x}_i) \tag{12.74}$$

El problema anterior, denominado *problema primal* (PP), se formula como sigue

$$\min_{\mathbf{x}=(\mathbf{x}_i,\mathbf{x}_j)} f(\mathbf{x}) = \sum_{i=1}^{I} f_i(\mathbf{x}_i) \tag{12.75}$$

sujeto a

$$\begin{array}{rcl} \mathbf{s}_i(\mathbf{x}_i) & \leq & \mathbf{0}, i = 1, \ldots, I \\ \mathbf{s}_j(\mathbf{x}_j) & \leq & \mathbf{0}, j = 1, \ldots, J \\ \sum\limits_{i=1}^{I} \mathbf{h}_i(\mathbf{x}_i) + \sum\limits_{j=1}^{J} \mathbf{h}_j(\mathbf{x}_j) & = & \mathbf{H} \\ \sum\limits_{i=1}^{I} \mathbf{g}_i(\mathbf{x}_i) + \sum\limits_{j=1}^{J} \mathbf{g}_j(\mathbf{x}_j) & \leq & \mathbf{G} \end{array} \tag{12.76}$$

Las restricciones de carga se denominan restricciones de complicación o restricciones globales de este problema primal. Las restricciones de carga incluyen

restricciones de igualdad (restricciones de demanda) y de desigualdad (restricciones de reserva de potencia). Estas restricciones acoplan las decisiones asociadas a las centrales térmicas e hidráulicas. Por ello, el problema anterior no puede descomponerse y resolverse fácilmente.

Mediante las técnicas de RL, las restricciones de carga se incorporan a la función objetivo para formar el problema primal relajado. El vector de multiplicadores asociado a las restricciones de demanda se denota mediante $\boldsymbol{\lambda}$, y el vector de multiplicadores asociado a las restricciones de reserva de potencia se denota mediante $\boldsymbol{\mu}$.

La función Lagrangiana se define como

$$\begin{array}{rcl}\mathcal{L}(\mathbf{x},\boldsymbol{\lambda},\boldsymbol{\mu}) & = & \sum\limits_{i=1}^{I} f_i(\mathbf{x}_i) + \boldsymbol{\lambda}^T\left(\mathbf{H} - \sum\limits_{i=1}^{I}\mathbf{h}_i(\mathbf{x}_i) - \sum\limits_{j=1}^{J}\mathbf{h}_j(\mathbf{x}_j)\right) \\ & & +\boldsymbol{\mu}^T\left(\mathbf{G} - \sum\limits_{i=1}^{I}\mathbf{g}_i(\mathbf{x}_i) - \sum\limits_{j=1}^{J}\mathbf{g}_j(\mathbf{x}_j)\right)\end{array} \tag{12.77}$$

y la función dual es la solución del problema

$$\theta(\boldsymbol{\lambda},\boldsymbol{\mu}) = \min_{(\mathbf{x}_i,\mathbf{x}_j)} \mathcal{L}(\mathbf{x}_i,\mathbf{x}_j,\boldsymbol{\lambda},\boldsymbol{\mu}) \tag{12.78}$$

sujeto a

$$\begin{array}{rcl}\mathbf{s}_i(\mathbf{x}_i) & \leq & \mathbf{0};\ \ i = 1,2,\ldots,I \\ \mathbf{s}_j(\mathbf{x}_j) & \leq & \mathbf{0};\ \ j = 1,2,\ldots,J\end{array} \tag{12.79}$$

que se puede expresar como

$$\theta(\boldsymbol{\lambda},\boldsymbol{\mu}) = \boldsymbol{\lambda}^T\mathbf{H} + \boldsymbol{\mu}^T\mathbf{G} + d(\boldsymbol{\lambda},\boldsymbol{\mu}) \tag{12.80}$$

donde $d(\boldsymbol{\lambda},\boldsymbol{\mu})$ es la solución del problema de optimización:

$$\min_{\mathbf{x}_i,\mathbf{x}_j}\left(\sum_{i=1}^{I}\Big(f_i(\mathbf{x}_i) - \boldsymbol{\lambda}^T\mathbf{h}_i(\mathbf{x}_i) - \boldsymbol{\mu}^T\mathbf{g}_i(\mathbf{x}_i)\Big) - \sum_{j=1}^{J}\Big(\boldsymbol{\lambda}^T\mathbf{h}_j(\mathbf{x}_j) + \boldsymbol{\mu}^T\mathbf{g}_j(\mathbf{x}_j)\Big)\right) \tag{12.81}$$

sujeto a

$$\begin{array}{c}\mathbf{s}_i(\mathbf{x}_i) \leq \mathbf{0}, i = 1,\ldots,I \\ \mathbf{s}_j(\mathbf{x}_j) \leq \mathbf{0}, j = 1,\ldots,J.\end{array} \tag{12.82}$$

Este problema se puede descomponer sistemáticamente en un subproblema por cada central térmica i y en un subproblema por cada central hidráulica j. Para valores fijos de $\boldsymbol{\lambda}$ y $\boldsymbol{\mu}$, el problema (12.78)-(12.79) se denomina *problema primal relajado*, y el problema (12.81)-(12.82) se denomina *problema primal descompuesto*.

El subproblema correspondiente a la central térmica i es

$$\min_{\mathbf{x}_i} f_i(\mathbf{x}_i) - \boldsymbol{\lambda}^T\mathbf{h}_i(\mathbf{x}_i) - \boldsymbol{\mu}^T\mathbf{g}_i(\mathbf{x}_i) \tag{12.83}$$

sujeto a

$$\mathbf{s}_i(\mathbf{x}_i) \leq \mathbf{0} \tag{12.84}$$

y el subproblema correspondiente a la central hidráulica j es

$$\max_{\mathbf{x}_j} \boldsymbol{\lambda}^T \mathbf{h}_j(\mathbf{x}_j) + \boldsymbol{\mu}^T \mathbf{g}_j(\mathbf{x}_j) \tag{12.85}$$

sujeto a

$$\mathbf{s}_j(\mathbf{x}_j) \leq \mathbf{0} \tag{12.86}$$

Las técnicas RL se basan en la solución del problema dual (el problema (9.64)):

$$\max_{\boldsymbol{\lambda},\boldsymbol{\mu}} \theta(\boldsymbol{\lambda}, \boldsymbol{\mu}) \tag{12.87}$$

sujeto a

$$\boldsymbol{\mu} \geq \mathbf{0} \tag{12.88}$$

Debido a la presencia de variables enteras (como las necesarias para representar el estado de acoplamiento de las centrales térmicas) en la formulación del problema primal, la CHTCP es un problema no convexo. Por tanto, la solución óptima del problema dual no es solución óptima del primal, sino una cota inferior. Sin embargo, a medida que aumenta el tamaño del problema, la holgura de dualidad unitaria se reduce (véase Ferreira [37], Bertsekas et al. [12] y Everett [34], por ejemplo), de forma que la solución óptima del problema dual es muy próxima a la solución óptima del problema primal. Una vez que se ha obtenido la solución óptima del problema dual, se emplean procedimientos heurísticos para obtener una buena solución del problema primal (Zhuang y Galiana [107]).

En el problema de la CHTCP, las restricciones de desigualdad se asocian a las variables enteras (estado de acoplamiento de las centrales térmicas) y las restricciones de igualdad se asocian a las variables continuas (potencia generada por las centrales térmicas e hidráulicas). Esta asociación hace que la búsqueda de una buena solución primal factible se divida en dos etapas. En la primera de ellas se modifica ligeramente la solución del problema dual, de forma que los valores que toman las variables enteras permitan el cumplimiento de las restricciones globales de desigualdad (restricciones de reserva de potencia). En la segunda etapa se modifica la solución de la etapa anterior ajustando los valores que toman las variables continuas, de forma que se cumplan las restricciones globales de igualdad (restricciones de demanda).

Por tanto, el algoritmo de la RL que resuelve el problema CHTCP consta de los siguientes pasos

Fase 1. Resolución del problema dual.

Fase 2A. Búsqueda de una solución primal factible en reserva de potencia.

Fase 2B. Búsqueda de una solución primal factible en demanda de potencia, que se llama despacho económico multiperiodo.

La resolución del problema dual (fase 1) es el paso más importante en los algoritmos que resuelven el problema de la CHTCP mediante RL. La eficiencia de estos algoritmos se mide según la eficiencia de esta fase.

La fase 2A es un procedimiento iterativo en el que los multiplicadores $\boldsymbol{\mu}$ se actualizan en aquellos períodos para los que no se cumple la reserva de potencia. En estos períodos, los multiplicadores $\boldsymbol{\mu}$ se incrementan proporcionalmente al incumplimiento de las restricciones de reserva de potencia (actualización tipo subgradiente) hasta que estas restricciones se cumplen para todos los períodos del horizonte temporal. Al final de la fase 2A, se obtiene una solución primal factible que se corresponde con un conjunto de decisiones de acoplamiento. Esta fase requiere poco tiempo de cálculo para encontrar una solución próxima a la solución de la fase 1. La fase 2B es un procedimiento de despacho económico multiperiodo (véase, por ejemplo, Wood y Wollenberg [104]) en la que, una vez que se obtiene el estado de acoplamiento de las centrales térmicas en la fase 2A, se ajusta la potencia generada de las centrales acopladas de forma que se cumpla la demanda en todos los períodos de tiempo. Este problema es el que resuelven diariamente los operadores de los sistemas eléctricos.

12.7.2 Resolución del problema dual

En el procedimiento de resolución del problema dual de la CHTCP se resuelve en cada iteración el problema primal relajado, y con la información obtenida, se actualizan los multiplicadores de Lagrange. La información que se obtiene al resolver el problema primal relajado es

- El valor de la función dual

$$\theta(\boldsymbol{\lambda}^{(t)}, \boldsymbol{\mu}^{(t)}) = \boldsymbol{\lambda}^{(t)T}\mathbf{H} + \boldsymbol{\mu}^{(t)T}\mathbf{G} + d(\boldsymbol{\lambda}^{(t)}, \boldsymbol{\mu}^{(t)}) \quad (12.89)$$

  donde

$$\begin{aligned} d(\boldsymbol{\lambda}^{(t)}, \boldsymbol{\mu}^{(t)}) = & \sum_{i=1}^{I}\Big(f_i(\mathbf{x}_i^{*(t)}) - \boldsymbol{\lambda}^{(t)T}\mathbf{h}_i(\mathbf{x}_i^{*(t)}) - \boldsymbol{\mu}^{(t)T}\mathbf{g}_i(\mathbf{x}_i^{*(t)})\Big) \\ & - \sum_{j=1}^{J}\Big(\boldsymbol{\lambda}^{(t)T}\mathbf{h}_j(\mathbf{x}_j^{*(t)}) + \boldsymbol{\mu}^{(t)T}\mathbf{g}_j(\mathbf{x}_j^{*(t)})\Big) \end{aligned} \quad (12.90)$$

  donde $\mathbf{x}_i^{*(t)}$ es el vector de los valores óptimos de las variables asociadas a la central térmica i y $\mathbf{x}_j^{*(t)}$ es el vector de los valores óptimos de las variables asociadas a la central hidráulica j, que se obtienen la resolver el problema primal relajado

- Un subgradiente $\mathbf{s}^{(t)}$ de la función dual en la solución óptima del problema relajado: $\mathbf{x}_i^*, \forall i$, $\mathbf{x}_j^*, \forall j$

Un subgradiente se calcula sencillamente; es el vector de incumplimientos de las restricciones de demanda junto al vector de incumplimiento de las restric-

ciones de reserva rodante:

$$\mathbf{s}^{(t)} = \text{columna}[\mathbf{h}^{(t)}, \mathbf{g}^{(t)}] \tag{12.91}$$

donde

$$\begin{aligned} \mathbf{h}^{(t)} &= \mathbf{H} - \sum_{i=1}^{I} \mathbf{h}_i(\mathbf{x}_i^{*(t)}) - \sum_{j=1}^{J} \mathbf{h}_j(\mathbf{x}_j^{*(t)}) \\ \mathbf{g}^{(t)} &= \mathbf{G} - \sum_{i=1}^{I} \mathbf{g}_i(\mathbf{x}_i^{*(t)}) - \sum_{j=1}^{J} \mathbf{g}_j(\mathbf{x}_j^{*(t)}) \end{aligned} \tag{12.92}$$

Entre los métodos de actualización de los multiplicadores de Lagrange, de interés para problemas de CHTCP, cabe destacar el método del subgradiente, los métodos de penalización (Pellegrino et al. [85]) y los métodos de los planos reconstructores (Jiménez y Conejo [60]).

12.7.3 Significado económico de los multiplicadores

Como se ha dicho anteriormente, una de las ventajas de emplear las técnicas de RL es que se dispone de información económica útil que se corresponde con las variables del problema dual, los multiplicadores de Lagrange.

El multiplicador $\boldsymbol{\lambda}$ para un período determinado, representa, desde la perspectiva del sistema, el coste de producir una unidad adicional de energía eléctrica (MWh), es decir, es el coste marginal de la energía eléctrica. Análogamente, desde la perspectiva de una compañía generadora, el multiplicador $\boldsymbol{\lambda}$, para un período dado, indica el ingreso marginal que una central debería recibir por cada MWh de energía. También indica el precio al que una compañía generadora podría ofertar en un mercado competitivo.

De igual manera, el multiplicador $\boldsymbol{\mu}$, para un período dado, representa el coste de disponibilidad para incrementar la potencia de una central o, en otras palabras, indica el ingreso marginal que debe recibir una compañía por cada MW que tiene disponible.

Esta interpretación económica resulta útil en los sistemas de energía eléctrica centralizados para confeccionar las tarifas, pero también es útil en sistemas eléctricos basados en mercados de energía.

En el marco de los sistemas eléctricos competitivos, el procedimiento de la RL que resuelve el problema de la CHTCP se puede considerar un mecanismo de determinación de precios en un mercado liberalizado. Es decir, se puede considerar como el mecanismo que permite satisfacer la demanda de los clientes cumpliendo ciertos niveles de seguridad (establecidos por medio de la reserva de potencia) y que elige las ofertas más baratas de los generadores.

El operador del sistema establece una propuesta de los precios horarios de la energía (multiplicadores de Lagrange) para todo el horizonte temporal. Cada generador (o compañía generadora) planifica su producción independientemente para el horizonte temporal considerado, de forma que sus beneficios sean máximos (es decir, cada generador resuelve un problema de optimización). Análogamente, cada sistema hidráulico planifica su producción de forma que obtenga el máximo beneficio posible (es decir, cada sistema hidráulico resuelve

un problema de optimización). Una vez que los generadores envían sus propuestas de producción, se evalúa la restricción de demanda en cada hora del horizonte temporal. El operador del mercado actualiza los precios horarios con alguna de las técnicas de actualización de multiplicadores, y se repite el proceso hasta que se cumple la demanda. Este proceso representa un mercado de energía competitivo. De igual manera, se procede para los mercados de reserva de potencia.

Es importante destacar que cuando se aplica las técnicas de RL al problema de la CHTCP, cada generador planifica su producción según los precios de la energía y de la reserva. El intercambio de información entre el operador del mercado y los generadores es claro y sencillo. El mercado que resulta es, por tanto, transparente y eficiente en términos económicos.

También es posible aplicar técnicas de descomposición basadas en el Lagrangiano aumentado para resolver el problema CHTCP. En Batut et al. [7], Batut y Renaud [8], Renaud [94], y Wang et al. [101] puede encontrarse más información sobre estas técnicas alternativas.

Capítulo 13

Algunos trucos útiles

13.1 Introducción

Este capítulo está organizado en dos partes. La primera presenta algunos trucos genéricos que son útiles para problemas de programación lineal y lineal entera-mixta. En la segunda parte se sugieren algunos trucos útiles para la formulación de problemas en GAMS.

13.2 Algunos trucos genéricos

En este apartado se presentan algunos trucos que facilitan, por un lado, la formulación y resolución de gran variedad de problemas, y por otro, permiten plantear tanto problemas no lineales como problemas lineales. En particular, se describen trucos útiles para los siguientes casos:

1. Tratamiento de variables no acotadas que por necesidades de programación deben formularse como no negativas
2. Conversión de un conjunto de restricciones lineales de desigualdad en un conjunto equivalente de restricciones lineales de igualdad
3. Conversión de un conjunto de restricciones lineales de igualdad en un conjunto equivalente de restricciones lineales de desigualdad
4. Conversión de un problema de maximización en un problema de minimización
5. Conversión de una función objetivo no lineal en su equivalente lineal
6. Tratamiento de algunas funciones no lineales como lineales
7. Tratamiento de un espacio lineal como un cono
8. Tratamiento de conjuntos alternativos de restricciones

9. Tratamiento de restricciones condicionales
10. Tratamiento de funciones no continuas
11. Tratamiento de funciones no convexas a trozos

13.2.1 Tratamiento de variables no acotadas

Cualquier conjunto de variables no acotadas $\{x_1, x_2, \ldots, x_r\}$ se puede sustituir por otro conjunto formado por la diferencia entre dos variables no negativas

$$\{y_1 - z_1, y_2 - z_2, \ldots, y_r - z_r | y_1, y_2, \ldots, y_r \geq 0; z_1, z_2, \ldots, z_r \geq 0\}$$

Esta sustitución implica que se duplica el número de variables del conjunto inicial.

Para hacer la anterior transformación se considera que la variable x_i tiene una parte positiva y otra negativa, que se definen como

$$\begin{array}{rclcl} y_i & = & x_i^+ & = & \max\{0, x_i\} \\ z_i & = & x_i^- & = & \max\{0, -x_i\} \end{array} \qquad (13.1)$$

respectivamente. Así, resulta sencillo comprobar que $x = x_i^+ - x_i^-$, donde tanto x_i^+ como x_i^- son no negativos. Sin embargo, como también se puede escribir $x = (x_i^+ + k) - (x_i^- + k)$ para cualquier $k > 0$, se tiene un número infinito de descomposiciones.

Una mejora al planteamiento anterior consiste en sustituir el conjunto las r variables no acotadas $\{x_1, \ldots, x_r\}$ por el conjunto $r+1$ de variables no negativas $\{x_1^*, \ldots, x_r^*, x^*\}$, donde

$$x_i \quad = \quad x_i^* - x^*; \quad i = 1, 2, \ldots, r \qquad (13.2)$$

De esta forma se añade una sola variable en lugar de las r que se añaden con el primer método.

El lector puede comprobar sin dificultad que si se elige

$$x^* = -\min\{x_1, x_2, \ldots, x_r\}; \quad x_i^* = x_i + x^*$$

entonces las variables $\{x_1^*, \ldots, x_r^*, x^*\}$ son no negativas. Nótese que si el conjunto $\{x_1^*, \ldots, x_r^*, x^*\}$ cumple la igualdad (13.2), entonces $\{x_1^*+k, \ldots, x_r^*+k, x^*+k\}$ también la cumple, para cualquier $k > 0$.

Ejemplo 13.1 (conversión a variables no negativas). Supóngase el siguiente conjunto de restricciones

$$\begin{array}{lllcr} x_1 & +x_2 & +x_3 & = & 1 \\ x_1 & -x_2 & -x_3 & \leq & 1 \\ x_1 & & +x_3 & \geq & -1 \\ x_1 & & & \geq & 0 \end{array} \qquad (13.3)$$

donde x_1 es la única variable no negativa. Si se necesita obtener un conjunto equivalente donde todas las variables sean no negativas, se tiene la siguiente transformación

$$\begin{array}{rcl} x_2 & = & y_2 - z_2 \\ x_3 & = & y_3 - z_3 \\ y_2, y_3, z_2, z_3 & \geq & 0 \end{array} \tag{13.4}$$

donde $y_2 = x_2^+$, $y_3 = x_3^+$, y $z_2 = x_2^-$, $z_3 = x_3^-$. Por tanto, el conjunto inicial de restricciones se sustituye por

$$\begin{array}{rrrrrcr} x_1 & +y_2 & -z_2 & +y_3 & -z_3 & = & 1 \\ x_1 & -y_2 & +z_2 & -y_3 & +z_3 & \leq & 1 \\ x_1 & & & +y_3 & -z_3 & \geq & -1 \\ x_1, & y_2, & z_2, & y_3, & z_3 & \geq & 0 \end{array} \tag{13.5}$$

■

Ejemplo 13.2 (conversión a variables no positivas: método alternativo). Una mejora respecto al método empleado en el ejemplo anterior consiste en realizar el siguiente cambio de variables:

$$\begin{array}{rcl} x_2 & = & y_2 - y \\ x_3 & = & y_3 - y \end{array} \tag{13.6}$$

y por tanto, el conjunto inicial de restricciones se sustituye por:

$$\begin{array}{rrrrcr} x_1 & +y_2 & +y_3 & -2y & = & 1 \\ x_1 & -y_2 & -y_3 & +2y & \leq & 1 \\ x_1 & & +y_3 & -y & \geq & -1 \\ x_1, & y_2, & y_3, & y, & \geq & 0 \end{array} \tag{13.7}$$

que contiene sólo una variable más que el conjunto (13.5). ■

13.2.2 Transformación de desigualdades en igualdades

Las restricciones de desigualdad pueden convertirse en restricciones de igualdad añadiendo lo que se conoce como *variables de holgura*:

- Si
$$a_{i1}x_1 + a_{i2}x_2 + \cdots + a_{in}x_n \leq b_i$$
entonces existe una variable $x_{n+1} \geq 0$ de forma que
$$a_{i1}x_1 + a_{i2}x_2 + \cdots + a_{in}x_n + x_{n+1} = b_i$$

- Si
$$a_{i1}x_1 + a_{i2}x_2 + \cdots + a_{in}x_n \geq b_i$$
entonces existe una variable $x_{n+1} \geq 0$ de forma que
$$a_{i1}x_1 + a_{i2}x_2 + \cdots + a_{in}x_n - x_{n+1} = b_i$$

Ejemplo 13.3 (transformación de desigualdades en igualdades). Si se introducen las variables de holgura u_1 y u_2, entonces las restricciones de desigualdad del conjunto de restricciones (13.5) se transforman en el siguiente conjunto de restricciones de igualdad:

$$\begin{array}{cccccccccc} x_1 & -y_2 & +z_2 & -y_3 & +z_3 & +u_1 & & = & 1 \\ x_1 & & & +y_3 & -z_3 & & -u_2 & = & -1 \end{array} \tag{13.8}$$

donde ahora se tiene que

$$x_1, y_2, y_3, z_2, z_3, u_1, u_2 \geq 0$$

■

13.2.3 Transformación de igualdades en desigualdades

Un conjunto de m igualdades siempre puede transformarse en un conjunto equivalente de $m+1$ desigualdades, como se demuestra en la siguiente proposición.

Proposición 13.1 (transformación de igualdades en desigualdades). *El conjunto de igualdades*

$$\mathbf{a}_i^T \mathbf{x} = b_i \tag{13.9}$$

es equivalente al conjunto de desigualdades

$$\begin{array}{rcl} \mathbf{a}_i^T \mathbf{x} & \leq & b_i, \; i = 1, \ldots, m \\ \left(\sum\limits_{i=1}^{m} \mathbf{a}_i^T \right) \mathbf{x} & \geq & \sum\limits_{i=1}^{m} b_i \end{array} \tag{13.10}$$

■

Demostración. Es claro que si se cumple (13.9) también se cumple (13.10). Inversamente, considérese que $k \in \{1, \ldots, m\}$. Entonces

$$\mathbf{a}_k^T \mathbf{x} = \left(\sum_{i=1}^{m} \mathbf{a}_i^T \right) \mathbf{x} - \left(\sum_{i \neq k} \mathbf{a}_i^T \right) \mathbf{x} \geq \sum_{i=1}^{m} b_i - \sum_{i \neq k} b_i = b_k$$

y teniendo en cuenta que $\mathbf{a}_k^T \mathbf{x} \leq b_k$, entonces

$$\mathbf{a}_k^T \mathbf{x} = b_k$$

■

Ejemplo 13.4 (transformación de igualdades en desigualdades). El conjunto de igualdades

$$\begin{array}{ccccc} x_1 & +x_2 & +x_3 & = & 0 \\ x_1 & -x_2 & -x_3 & = & 2 \\ x_1 & & +x_3 & = & -1 \end{array}$$

es equivalente al conjunto de desigualdades

$$\begin{array}{rrrcr} x_1 & +x_2 & +x_3 & \leq & 0 \\ x_1 & -x_2 & -x_3 & \leq & 2 \\ x_1 & & +x_3 & \leq & -1 \\ 3x_1 & & +x_3 & \geq & 1 \end{array}$$

■

13.2.4 Transformación de maximización a minimización

Cualquier problema de maximización equivale a uno de minimización cuya función objetivo tiene signo contrario. Concretamente, maximizar

$$Z_{max} = \mathbf{c}^T \mathbf{x}$$

equivale a minimizar

$$Z_{min} = -\mathbf{c}^T \mathbf{x}$$

donde ambos problemas están sujetos al mismo conjunto de restricciones. Obsérvese que $Z_{max} = -Z_{min}$, aunque el valor óptimo de las variables de ambos problemas coincide.

13.2.5 Transformación de funciones no lineales en lineales

El problema de minimización

$$Z = f(\mathbf{x})$$

sujeto a

$$\begin{array}{rcl} \mathbf{h}(\mathbf{x}) & = & \mathbf{0} \\ \mathbf{g}(\mathbf{x}) & \leq & \mathbf{0} \end{array} \qquad (13.11)$$

es equivalente al problema de minimización

$$Z = y$$

sujeto a

$$\begin{array}{rcl} \mathbf{h}(\mathbf{x}) & = & \mathbf{0} \\ \mathbf{g}(\mathbf{x}) & \leq & \mathbf{0} \\ f(\mathbf{x}) & \leq & y \end{array} \qquad (13.12)$$

La demostración de esta equivalencia es sencilla.

Este truco resulta útil para aquellas técnicas de resolución que requieran una función objetivo lineal.

13.2.6 Tratamiento de funciones no lineales como lineales

Los PPL se restringen a problemas cuya función objetivo es lineal y también lo son sus restricciones. Cuando la función objetivo o alguna de las restricciones no es lineal, se tiene un PPNL. Sin embargo, para ciertos casos, es posible transformar un PPNL en un PPL equivalente.

Tratamiento de valores absolutos

La función valor absoluto es una función que se utiliza con cierta frecuencia en problemas de optimización. Esta función, además de ser no lineal, es una función no diferenciable.

El PPNL que minimiza

$$Z = |\mathbf{c}^T\mathbf{x}|$$

sujeto a

$$\mathbf{A}\mathbf{x} = \mathbf{b}$$

equivale al PPL que minimiza

$$Z = y$$

sujeto a

$$\begin{array}{rcl} \mathbf{A}\mathbf{x} & = & \mathbf{b} \\ \mathbf{c}^T\mathbf{x} & \leq & y \\ -\mathbf{c}^T\mathbf{x} & \leq & y \\ y & \geq & 0 \end{array}$$

Genéricamente, el PPNL que minimiza

$$Z = \sum_{i=1}^{m} |\mathbf{c}_i^T\mathbf{x}|$$

sujeto a

$$\mathbf{A}\mathbf{x} = \mathbf{b}$$

equivale al PPL que minimiza

$$Z = \sum_{i=1}^{m} y_i$$

sujeto a

$$\begin{array}{rcll} \mathbf{A}\mathbf{x} & = & \mathbf{b} & \\ \mathbf{c}_i^T\mathbf{x} & \leq & y_i, & i = 1, 2, \ldots, m \\ -\mathbf{c}_i^T\mathbf{x} & \leq & y_i, & i = 1, 2, \ldots, m \\ y_i & \geq & 0, & i = 1, 2, \ldots, m \end{array}$$

Tratamiento de la función máximo

Otra función de interés es la función máximo que también es no lineal y no diferenciable. A continuación, se presenta un truco para el tratamiento de esta función.

El PPNL que minimiza

$$Z = \max_{i=1,2,\ldots,m} |\mathbf{c}_i^T\mathbf{x}|$$

sujeto a

$$\mathbf{Ax} = \mathbf{b}$$

equivale al PPL que minimiza

$$Z = y$$

sujeto a

$$\begin{array}{rcl} \mathbf{Ax} & = & \mathbf{b} \\ \mathbf{c}_i^T\mathbf{x} & \leq & y, \quad i = 1, 2, \dots, m \\ -\mathbf{c}_i^T\mathbf{x} & \leq & y, \quad i = 1, 2, \dots, m \\ y & \geq & 0 \end{array}$$

El Problema de la programación lineal fraccional

El *problema de la programación fraccional* consiste en optimizar el cociente de dos funciones lineales sujeto a un conjunto de restricciones también lineales. Este problema puede plantearse como sigue. Minimizar

$$\frac{\mathbf{p}^{\mathbf{T}}\mathbf{x} + \alpha}{\mathbf{q}^{\mathbf{T}}\mathbf{x} + \beta} \tag{13.13}$$

sujeto a

$$\begin{array}{rcl} \mathbf{Ax} & \leq & \mathbf{b} \\ \mathbf{x} & \geq & \mathbf{0} \end{array} \tag{13.14}$$

donde $\mathbf{p}$ y $\mathbf{q}$ son vectores de dimensión n, $\mathbf{b}$ es un vector de dimensión m, $\mathbf{A}$ es una matriz de dimensión $m \times n$, y α y β son escalares.

Este problema puede transformarse en un PPL mediante el siguiente teorema debido a Charnes y Cooper [25].

Teorema 13.1 (programación fraccional). *Si el conjunto factible*

$$\mathbf{X} = \{\mathbf{x} | \mathbf{Ax} \leq \mathbf{b} \text{ y } \mathbf{x} \geq \mathbf{0}\}$$

es acotado y no vacío, y si $\mathbf{q}^T\mathbf{x} + \beta > 0$ *para cada* $\mathbf{x} \in \mathbf{X}$*, el problema (13.13)–(13.14) equivale al PPL: Minimizar*

$$\mathbf{p}^T\mathbf{y} + \alpha z \tag{13.15}$$

sujeto a

$$\mathbf{Ay} - \mathbf{b}z \leq \mathbf{0} \tag{13.16}$$
$$\mathbf{q}^T\mathbf{y} + \beta z = 1 \tag{13.17}$$
$$\mathbf{y} \geq \mathbf{0} \tag{13.18}$$
$$z \geq 0 \tag{13.19}$$

que tiene sólo una variable y una restricción adicionales. ∎

Este teorema se basa en la siguiente trasformación:

$$z = \frac{1}{\mathbf{q}^T\mathbf{x} + \beta} \quad \text{and} \quad \mathbf{y} = z\mathbf{x}, \tag{13.20}$$

Ejemplo 13.5 (programación fraccional). Supóngase el siguiente problema de programación lineal fraccional. Minimizar

$$\frac{x_1 + 1}{x_2 + 2}$$

sujeto a

$$\begin{array}{rcl} x_1 + x_2 & \leq & 1 \\ x_1, x_2 & \geq & 0 \end{array}$$

Para transformar este problema de programación lineal fraccional en uno lineal, se realiza la transformación (13.20)

$$z = \frac{1}{x_2 + 1}; \ \mathbf{y} = z\mathbf{x},$$

y se obtiene el siguiente PPL. Minimizar

$$y_1 + z$$

sujeto a

$$\begin{array}{rcl} y_1 + y_2 - z & \leq & 0 \\ y_2 + 2z & = & 1 \\ y_1, y_2, z & \geq & 0 \end{array}$$

■

13.2.7 Espacio vectorial como cono

En ocasiones, puede resultar interesante tratar sólo con las componentes no negativas de un vector. La siguiente proposición puede ser útil en estos casos.

Proposición 13.2 (espacio lineal como cono). *Un espacio lineal* $\mathbf{A}_\rho$ *es un caso particular del cono*

$$\mathbf{A}_\rho \equiv (\mathbf{A} : -\mathbf{A})_\pi, \tag{13.21}$$

donde $-\mathbf{A}$ *es la parte negativa de* $\mathbf{A}$. *En otras palabras, un espacio lineal es el cono generado por sus generadores y los vectores opuestos de los mismos.* ■

Obsérvese que esto equivale a duplicar el número de generadores. Sin embargo, existe una solución más eficiente que la que muestra el siguiente teorema.

Teorema 13.2 (espacio vectorial como cono). *Dado un espacio lineal* $\mathbf{A}_\rho$ *y un vector* $\mathbf{x} \in \mathbf{A}_\rho$ *con todas sus componentes positivas respecto al conjunto de generadores, entonces* $\mathbf{A}_\rho \equiv (\mathbf{A} : -\mathbf{x})_\pi$. ■

Demostración. Como $(\mathbf{A} : -\mathbf{x})_\pi \subseteq \mathbf{A}_\rho$, sólo se necesita demostrar que $\mathbf{A}_\rho \subseteq (\mathbf{A} : -\mathbf{x})_\pi$. Puesto que $\mathbf{x}$ puede expresarse como:

$$\mathbf{x} = \sum_{i=1}^{m} \sigma_i \mathbf{a}_i; \quad \sigma_i > 0; i = 1, \ldots, m \tag{13.22}$$

si $\mathbf{y} \in \mathbf{A}_\rho$, para cualquier real π, se tiene

$$\mathbf{y} = \sum_{i=1}^{m} \rho_i \mathbf{a}_i = \sum_{i=1}^{m} \rho_i \mathbf{a}_i + \pi\mathbf{x} - \pi\mathbf{x} = \sum_{i=1}^{m} (\rho_i + \pi\sigma_i)\, \mathbf{a}_i + \pi(-\mathbf{x}) \tag{13.23}$$

Si se elige

$$\pi = \max_{i=1,\ldots,m} \left| \frac{\rho_i}{\sigma_i} \right| \geq 0 \tag{13.24}$$

se tiene $\rho_i + \pi\sigma_i \geq 0; i = 1, \ldots, m$, y finalmente $\mathbf{y} \in (\mathbf{A} : -\mathbf{x})_\pi$. ■

Nota 13.1 *El teorema anterior implica en la práctica que cualquier espacio lineal generado por m vectores puede considerarse como un cono generado por* $m + 1$ *vectores. Esto mejora el resultado de la proposición 13.2, expresión (13.21), que requiere* $2m$ *generadores.*

Para expresar un espacio lineal como un cono, se puede elegir $\mathbf{x}$ *de muchas formas, sin embargo,* $\mathbf{x} = \sum_{i=1}^{m} \mathbf{a}_i$, *conduce a* $\pi = \max_{i=1,\ldots,m} |\rho_i|$, *que es una elección muy conveniente de* $\mathbf{x}$. ■

Ejemplo 13.6 (espacio vectorial como un cono). Considérese el espacio vectorial $\mathbf{A}_\rho$ donde

$$\mathbf{A} = \begin{pmatrix} 1 & -1 \\ 2 & 0 \\ 3 & 1 \end{pmatrix}$$

De acuerdo con la nota 13.1, se puede aplicar el teorema 13.2, con

$$\mathbf{x} = \sum_{i=1,2} \mathbf{a}_i = (1, 2, 3)^T + (-1, 0, 1)^T = (0, 2, 4)^T$$

y obtener $\mathbf{A}_\rho \equiv (\mathbf{A} : -\mathbf{x})_\pi \equiv \mathbf{B}_\pi$, con

$$\mathbf{B} = \begin{pmatrix} 1 & -1 & 0 \\ 2 & 0 & -2 \\ 3 & 1 & -4 \end{pmatrix}$$

Ahora bien, dado el vector $\mathbf{y}$, de $\mathbf{A}_\rho$, como

$$\mathbf{y} = -(1, 2, 3)^T - 2(-1, 0, 1)^T = (1, -2, -5)^T$$

se puede elegir

$$\pi = \max_{i=1,2} |\rho_i| = \max(|-1|, |-2|) = 2$$

y luego

$$\begin{array}{rcl} \mathbf{y} & = & (1,-2,-5)^T = (-1+2)(1,2,3)^T + (-2+2)(-1,0,1)^T + 2(0,-2,-4)^T \\ & = & (1,2,3)^T + 2(0,-2,-4)^T = \mathbf{a}_1 + 2(-\mathbf{x}) \end{array} \quad (13.25)$$

que es una combinación lineal no negativa del vector $\mathbf{a}_1$ y del vector $-\mathbf{x}$. ∎

13.2.8 Restricciones alternativas

Las variables binarias constituyen una poderosa herramienta para modelar no linealidades de diversa naturaleza, que aparecen con frecuencia en el mundo de la ingeniería. Una aplicación posible de las variables binarias es modelar el cumplimiento de al menos uno de dos conjuntos de restricciones. Para conseguirlo, se necesita la siguiente proposición.

Proposición 13.3 (restricciones alternativas). *Considérense dos conjuntos de restricciones:*

$$\mathbf{A}_1^T\mathbf{x} \leq \mathbf{b}_1 \quad (13.26)$$

$$\mathbf{A}_2^T\mathbf{x} \leq \mathbf{b}_2 \quad (13.27)$$

El conjunto de restricciones que asegura el cumplimiento de al menos una de las restricciones anteriores es:

$$\mathbf{A}_1^T\mathbf{x} - y_1\mathbf{d}_1 \leq \mathbf{b}_1 \quad (13.28)$$

$$\mathbf{A}_1^T\mathbf{x} - y_2\mathbf{d}_2 \leq \mathbf{b}_2 \quad (13.29)$$

$$y_1 + y_2 \leq 1 \quad (13.30)$$

$$y_1, y_2 \in \{0,1\} \quad (13.31)$$

Las matrices columna $\mathbf{d}_1$ y $\mathbf{d}_2$ deben satisfacer para todo $\mathbf{x}$ que:

$$\mathbf{A}_1^T\mathbf{x} \leq \mathbf{b}_1 + \mathbf{d}_1 \Leftrightarrow \mathbf{d}_1 \geq \mathbf{A}_1^T\mathbf{x} - \mathbf{b}_1 \quad (13.32)$$

$$\mathbf{A}_2^T\mathbf{x} \leq \mathbf{b}_2 + \mathbf{d}_2 \Leftrightarrow \mathbf{d}_2 \geq \mathbf{A}_2^T\mathbf{x} - \mathbf{b}_2. \quad (13.33)$$

Por otro lado, si se necesita que se cumpla tan sólo una restricción de las dos posibles, entonces la condición $y_1 + y_2 \leq 1$ se debe sustituir por $y_1 + y_2 = 1$. ∎

Demostración. Puesto que se cumplen (13.30) y (13.31), se tiene tres casos posibles.

Caso 1. $y_1 = y_2 = 0$. En este caso, (13.28) y (13.29) se transforman en (13.26) y (13.27), respectivamente; por tanto se cumplen las dos restricciones.

Caso 2. $y_1 = 0; \;\; y_2 = 1$. Puesto que (13.33) se cumple para todo x, y como (13.29) se transforma en (13.33), (13.29) no está activa, es decir, se puede eliminar.

Caso 3. $y_1 = 1; \;\; y_2 = 0$. Puesto que (13.32) se cumple para todo x, y como (13.28) se transforma en (13.32), (13.28) no está activa, es decir, se puede eliminar.

■

Ejemplo 13.7 (restricciones alternativas). Considérese el siguiente conjunto de restricciones:

$$\begin{array}{rcl} -x_1 & \leq & 0 \\ x_2 & \leq & 2 \\ 2x_1 - x_2 & \leq & 0 \end{array} \Leftrightarrow \begin{pmatrix} -1 & 0 \\ 0 & 1 \\ 2 & -1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \leq \begin{pmatrix} 0 \\ 2 \\ 0 \end{pmatrix} \tag{13.34}$$

y

$$\begin{array}{rcl} -x_1 & \leq & 0 \\ -x_2 & \leq & 0 \\ x_1 + x_2 & \leq & 1 \end{array} \Leftrightarrow \begin{pmatrix} -1 & 0 \\ 0 & -1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \leq \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \tag{13.35}$$

que delimitan los conjuntos factible A y B, que aparecen en la figura 13.1 con sombreado claro y oscuro, respectivamente.

Para calcular los límites correspondientes a (13.32) y (13.33) se puede buscar aquella región que con las menos modificaciones posibles tenga unos límites paralelos a los límites de A que contienen a B. Esta región es OST. Análogamente, aquella región que con las menos modificaciones posibles tenga sus límites paralelos a los límites de B y que contiene a A, es la región PQR.

Estas dos nuevas regiones pueden definirse mediante el siguiente conjunto de restricciones

$$\begin{array}{rcl} -x_1 & \leq & 0 \\ x_2 & \leq & 2 \\ 2x_1 - x_2 & \leq & 2 \end{array} \Leftrightarrow \begin{pmatrix} -1 & 0 \\ 0 & 1 \\ 2 & -1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} - y_1 \begin{pmatrix} 0 \\ 0 \\ 2 \end{pmatrix} \leq \begin{pmatrix} 0 \\ 2 \\ 0 \end{pmatrix} \tag{13.36}$$

$$\begin{array}{rcl} -x_1 & \leq & 0 \\ -x_2 & \leq & 0 \\ x_1 + x_2 & \leq & 3 \end{array} \Leftrightarrow \begin{pmatrix} -1 & 0 \\ 0 & 1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} - y_2 \begin{pmatrix} 0 \\ 0 \\ 2 \end{pmatrix} \leq \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \tag{13.37}$$

de lo que se obtiene que

$$\mathbf{d}_1 = \mathbf{d}_2 = \begin{pmatrix} 0 \\ 0 \\ 2 \end{pmatrix} \tag{13.38}$$

Sin embargo, basta con elegir $\mathbf{d}_1$ y $\mathbf{d}_2$ suficientemente grandes.

■

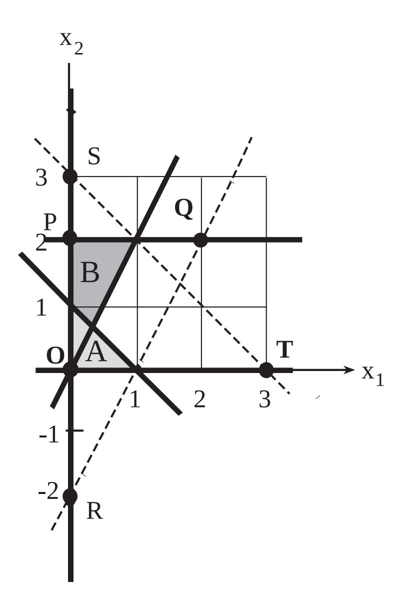


Figura 13.1: Representación gráfica de las restricciones alternativas del ejemplo 13.7.

13.2.9 Tratamiento de restricciones condicionales

Una condición del tipo

$$f_1(x_1, \ldots, x_n) > b_1 \qquad \text{implica que} \qquad f_2(x_1, \ldots, x_n) \leq b_2 \tag{13.39}$$

es equivalente al siguiente conjunto de restricciones

$$f_1(x_1, \ldots, x_n) \leq b_1 \qquad \text{y/o} \qquad f_2(x_1, \ldots, x_n) \leq b_2 \tag{13.40}$$

Las restricciones alternativas se analizaron en el apartado 13.2.8.

A continuación se demuestra la equivalencia anterior. El conjunto original de condiciones únicamente se incumple cuando

$$f_1(x_1, \ldots, x_n) > b_1 \qquad \text{y} \qquad f_2(x_1, \ldots, x_n) > b_2 \tag{13.41}$$

y, por tanto, se cumple cuando

$$f_1(x_1, \ldots, x_n) \leq b_1 \qquad \text{y/o} \qquad f_2(x_1, \ldots, x_n) \leq b_2 \tag{13.42}$$

13.2.10 Tratamiento de funciones no continuas

Las variables binarias también permiten el tratamiento de funciones no continuas, lo que se demuestra en la siguiente proposición.

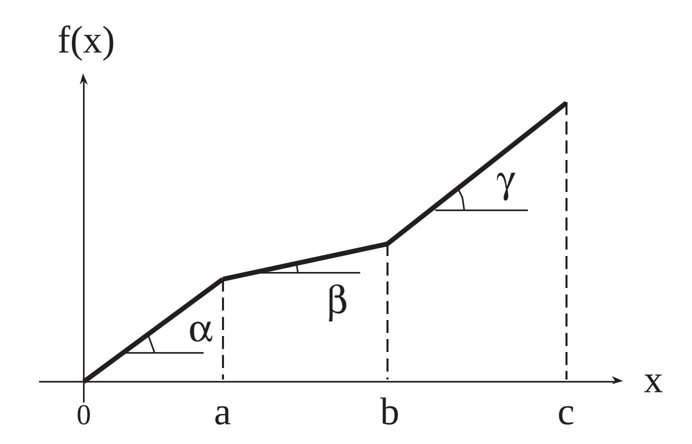


Figura 13.2: Función no convexa a trozos.

Proposición 13.4 (discontinuidad). *La función discontinua a minimizar*

$$f(x) = \begin{cases} 0 & x = 0 \\ k + cx & 0 < x \leq b; \; k > 0 \end{cases} \tag{13.43}$$

se puede formular como

$$\begin{array}{rcl} f(x) & = & ky + cx \\ x & \leq & by \\ -x & \leq & 0 \\ y & \in & \{0, 1\} \end{array} \tag{13.44}$$

■

Demostración. Se analizan las dos únicas posibilidades:

Caso 1. Si $y = 0$ por la segunda y tercera ecuación en (13.44), se tiene que $0 \leq x \leq 0$; es decir, $x = 0$, y sustituyendo $x = y = 0$ en la primera ecuación se tiene que $f(x) = 0$.

Caso 2. Si, por el contrario $y = 1$, se tiene que $0 \leq x \leq b$ y $f(x) = k + cx$. Obsérvese que la discontinuidad en cero $(f(0) = 0, f(0^+) = k)$ deja de existir al minimizar $f(x)$, pues esto implica que $f(0^-) = 0$. ■

13.2.11 Tratamiento de funciones no convexas a trozos

En este apartado se explica cómo usar variables binarias para tratar funciones no convexas a trozos.

Proposición 13.5 (función no convexa a trozos). *La función no convexa a trozos (véase la figura 13.2)*

$$f(x) = \begin{cases} \alpha x & 0 \le x \le a \\ \alpha a + \beta(x-a) & a < x \le b \\ \alpha a + \beta(b-a) + \gamma(x-b) & b < x \le c \end{cases}$$

donde

$$\beta < \alpha < \gamma; \quad \alpha, \beta, \gamma > 0; \quad a < b < c; \quad a, b, c > 0$$

puede expresarse como

$$\begin{aligned} f(x) &= \alpha x_1 + \beta x_2 + \gamma x_3 & (13.45) \\ x &= x_1 + x_2 + x_3 & (13.46) \\ a w_1 &\le x_1 \le a & (13.47) \\ w_2(b-a) &\le x_2 \le (b-a)w_1 & (13.48) \\ 0 &\le x_3 \le (c-b)w_2 & (13.49) \\ w_2 &\le w_1 & (13.50) \\ w_1, w_2 &\in \{0,1\} & (13.51) \end{aligned}$$

∎

Demostración. Como consecuencia de las dos últimas restricciones (13.50) y (13.51), se tiene tres casos posibles:

Caso 1. $w_2 = w_1 = 0$. Lo anterior, junto con (13.47)–(13.49), implica que $x_2 = x_3 = 0$ y $0 \le x_1 \le a$, y, por tanto, (13.46) lleva a $x = x_1$ y $0 \le x \le a$, y $f(x) = \alpha x$.

Caso 2. $w_2 = 0; \quad w_1 = 1$. Lo anterior, junto con (13.54)–(13.56), implica que $x_3 = 0$ y $0 \le x_2 \le b - a$ y $x_1 = a$, y, por tanto, (13.53) lleva a $x = a + x_2 \Rightarrow a \le x \le b$ y $f(x) = \alpha a + \beta(x-a)$.

Caso 3. $w_2 = w_1 = 1$. Lo anterior, junto con (13.54)–(13.56), implica que $0 \le x_3 \le c - b$, $x_2 = b - a$ y $x_1 = a$, y, por tanto, (13.53) lleva a $x = b + x_3 \Rightarrow b \le x \le c$ y $f(x) = \alpha a + \beta(b-a) + \gamma(x-b)$. ∎

Proposición 13.6 (función no convexa a trozos con discontinuidad inicial). *La función no convexa a trozos con discontinuidad inicial (véase la figura 13.3)*

$$f(x) = \begin{cases} 0 & 0 \le x < a \\ f_0 + \alpha(x-a) & a < x \le b \\ f_0 + \alpha(b-a) + \beta(x-b) & b < x \le c \\ f_0 + \alpha(b-a) + \beta(c-b) + \gamma(x-c) & c < x \le d \end{cases}$$

donde

$$0 < \beta < \alpha < \gamma; \quad 0 < a < b < c < d$$

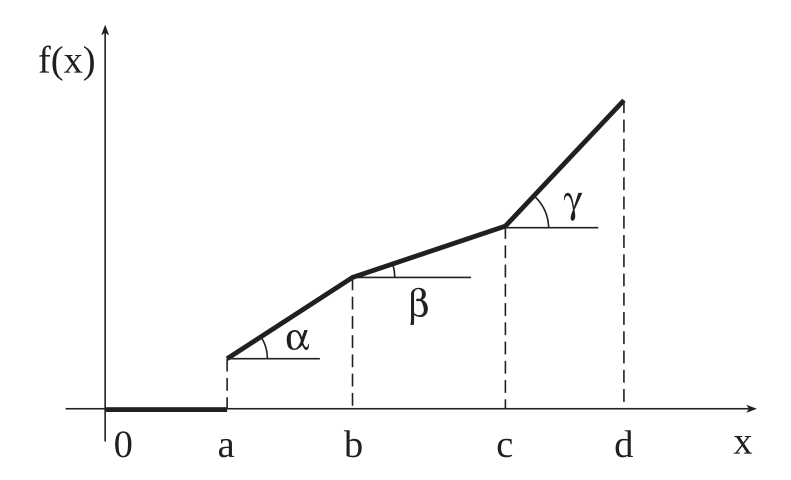


Figura 13.3: Función no convexa a trozos con discontinuidad inicial.

puede expresarse como

$$
\begin{aligned}
f(x) &= vf_0 + \alpha x_1 + \beta x_2 + \gamma x_3 && (13.52)\\
x &= va + x_1 + x_2 + x_3 && (13.53)\\
w_1(b-a) &\leq x_1 \leq (b-a)v && (13.54)\\
w_2(c-b) &\leq x_2 \leq (c-b)w_1 && (13.55)\\
0 &\leq x_3 \leq (d-c)w_2 && (13.56)\\
w_2 &\leq w_1 \leq v && (13.57)\\
v, w_1, w_2 &\in \{0,1\} && (13.58)
\end{aligned}
$$

■

Demostración. Como consecuencia de las dos últimas restricciones (13.57) y (13.58), se tiene cuatro casos posibles:

Caso 1. $w_2 = w_1 = v = 0$. Lo anterior, junto con (13.54)–(13.56), implica que $x_1 = x_2 = x_3 = 0$, y, por tanto, (13.53) lleva a $x = 0$ y $f(x) = 0$.

Caso 2. $w_2 = w_1 = 0$; $v = 1$. Lo anterior, junto con (13.54)–(13.56), implica que $x_2 = x_3 = 0$ y $0 \leq x_1 \leq b - a$, y por tanto, (13.53) lleva a $x = a + x_1 \Rightarrow a \leq x \leq b$ y $f(x) = f_0 + \alpha(x - a)$.

Caso 3. $w_2 = 0$; $w_1 = v = 1$. Lo anterior, junto con (13.54)–(13.56), implica que $x_3 = 0$, $0 \leq x_2 \leq c - b$ y $x_1 = b - a$, y por tanto (13.53) lleva a $x = b + x_2 \Rightarrow b \leq x \leq c$ y $f(x) = f_0 + \alpha(b - a) + \beta(x - b)$.

Caso 4. $w_2 = w_1 = v = 1$. Lo anterior, junto con (13.54)–(13.56), implica que $0 \leq x_3 \leq d - c$, $x_2 = c - b$ y $x_1 = b - a$, y por tanto (13.53) lleva a $x = c + x_3 \Rightarrow c \leq x \leq d$ y $f(x) = f_0 + \alpha(b - a) + \beta(c - b) + \gamma(x - c)$.

■

13.3 Algunos trucos en GAMS

En este apartado, se presentan los siguientes trucos en GAMS:

- Asignación de valores a una matriz
- Definición de matrices simétricas
- Definición de matrices cuasi-vacías
- Descomposición de un problema en subproblemas
- Adición iterativa de restricciones
- Tratamiento de los estados inicial y final
- Análisis de sensibilidad
- Dependencia del flujo del programa con respecto a la resolución de un problema

13.3.1 Asignación de valores a una matriz

La mayoría de los lenguajes de programación utilizan bucles para asignar valores a los elementos de una matriz. En GAMS, esto no es necesario, el usuario de GAMS puede asignar un valor inicial a todos los elementos de la matriz con una sola sentencia. Esto se debe a que en GAMS la asignación se hace en 'paralelo'. El siguiente ejemplo ilustra cómo realizar esta asignación. Sean las siguientes matrices:

$$\mathbf{m_1} = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}, \quad \mathbf{m_2} = \begin{pmatrix} 2 & 3 & 4 \\ 3 & 4 & 5 \end{pmatrix}$$

El código para definir ambas matrices puede ser el siguiente:

```
SETS   I  indice de filas    /1*2/
       J  indice de columnas /1*3/;

PARAMETER m1(I,J),m2(i,J);

** Asignacion en paralelo para todos los elementos
 m1(I,J)=1;
 m2(I,J)=ord(I)+ord(J);

DISPLAY m1,m2;
```

La anterior sentencia DISPLAY produce:

```
----       9 PARAMETER M1
             1            2            3
1        1.000        1.000        1.000
2        1.000        1.000        1.000

----       9 PARAMETER M2
             1            2            3
1        2.000        3.000        4.000
2        3.000        4.000        5.000
```

13.3.2 Definición de una matriz simétrica

Para representar una matriz simétrica basta con definir la submatriz triangular superior. Los valores de la submatriz triangular inferior se pueden asignar posteriormente utilizando una expresión condicional. Para ilustrar este truco, se considera la siguiente matriz simétrica:

$$\mathbf{a} = \begin{pmatrix} 1 & 0 & 3 \\ 0 & 2 & 4 \\ 3 & 4 & 5 \end{pmatrix}$$

El código GAMS que permite definir esta matriz es el siguiente:

```
SET   I  indice de filas y columnas  /1*3/
ALIAS(I,J);

PARAMETER a(I,J) matriz de datos
          /1.1  1
           2.2  2
           3.3  5
           1.3  3
           2.3  4
          /;

** Primero se muestra, en la salida, la submatriz triangular superior
DISPLAY a;

** Se asignan valores a la submatriz triangular inferior
a(I,J)$(ord(I) gt ord(J))=a(J,I);

** Se muestra en la salida la matriz completa
DISPLAY a;
```

El resultado que produce la sentencia DISPLAY es el siguiente:

```
----      13 PARAMETER A             matriz de datos
             1            2            3
1        1.000                     3.000
2                     2.000        4.000
3                                  5.000

----      19 PARAMETER A             matriz de datos
```

```
         1           2           3
1      1.000                   3.000
2                  2.000       4.000
3      3.000       4.000       5.000
```

En el apartado 11.4.10 se presenta un ejemplo alternativo.

13.3.3 Definición de una matriz cuasi-vacía

Las matrices cuasi-vacías se pueden definir de distintas maneras en GAMS. La opción más sencilla es definirlas por enumeración de los elementos distintos de cero. Esta opción es apropiada cuando la posición de los elementos distintos de cero no sigue ningún patrón determinado. Sin embargo, las matrices cuasi-vacías de gran dimensión suelen tener una distribución determinada de sus elementos distintos de cero. En estos casos, puede resultar ineficiente la enumeración de los elementos, y es mejor aprovechar el conocimiento de la distribución que siguen estos. Si la posición de los elementos distintos de cero sigue un patrón, las asignaciones pueden realizarse imponiendo condiciones sobre los índices de sus posiciones. De lo contrario, es necesario definir un conjunto especial que establezca la correspondencia entre elementos que tengan el mismo valor. Sea la siguiente matriz

$$\mathbf{s} = \begin{pmatrix} 0 & 0 & 1 & 2 & 0 & 0 \\ 0 & 1 & 2 & 0 & 0 & 1 \\ 1 & 2 & 0 & 0 & 1 & 0 \\ 2 & 0 & 0 & 1 & 0 & 2 \\ 0 & 0 & 1 & 0 & 2 & 0 \\ 0 & 1 & 0 & 2 & 0 & 1 \end{pmatrix}$$

Un código que aprovecha la distribución de los elementos distintos de cero puede ser

```
SET  I  indice de filas y columnas /I1*I6/;

ALIAS(I,J);

PARAMETER s(I,J) matriz cuasi-vacia;
 s(I,J)$(mod((ord(I)+ord(J)),4) eq 0)=1;
 s(I,J)$(mod((ord(I)+ord(J)),5) eq 0)=2;
DISPLAY s;
```

El siguiente código define la matriz **s** mediante la enumeración de las posiciones que ocupan los elementos distintos de cero.

```
SETS  I  indice de filas y columnas /I1*I6/
      MAP1(I,I)     /I1.I3,I2.I2,I2.I6,I3.I5,I4.I4,I6.I6/
      MAP2(I,I)     /I1.I4,I2.I3,I4.I6,I5.I5 /;
ALIAS(I,J);

PARAMETER s(I,J) matriz cuasi-vacia;

 s(I,J)$MAP1(I,J)=1;
```

```
 s(I,J)$MAP2(I,J)=2;

 s(I,J)$(ord(I) gt ord(J))=s(J,I);

DISPLAY s;
```

Los dos códigos anteriores producen la misma matriz:

```
----    10 PARAMETER S            matriz cuasi-vacia
            I1          I2          I3          I4          I5          I6
I1                               1.000       2.000
I2                   1.000       2.000                               1.000
I3       1.000       2.000                               1.000
I4       2.000                               1.000                   2.000
I5                               1.000                   2.000
I6                   1.000                   2.000                   1.000
```

13.3.4 Descomposición de un problema separable

En algunas ocasiones, desde una perspectiva computacional, puede resultar útil descomponer un problema en varios subproblemas. En estos casos, se puede formular el problema igual que si no fuese separable, y luego, mediante un bucle, se resuelven secuencialmente los distintos subproblemas. Por ejemplo, dado el siguiente problema de optimización. Minimizar

$$Z = \sum_{j=1}^{n} \sum_{i=1}^{n} c_{ij} x_{ij}$$

sujeto a

$$\sum_{i=1}^{n} a_i x_{ij} \geq b_j \quad j = 1, \ldots, n$$

Este problema se puede descomponer en tantos subproblemas como el cardinal del intervalo de variación del índice j (un total de n subproblemas). Cada uno de estos subproblemas tiene la forma siguiente: Minimizar

$$Z = \sum_{i=1}^{n} c_{ij} x_{ij}$$

sujeto a

$$\sum_{i=1}^{n} a_i x_{ij} \geq b_j$$

Para comprobar cómo GAMS resuelve los n subproblemas, se sugiere estudiar el fichero de salida GAMS que genera el siguiente código:

```
SETS   I  indice de filas    /1*2/
       J  indice de columnas /1*3/

       DIN(J) conjunto dinamico;

ALIAS(J,J1);
```

```
PARAMETERS c(I,J),a(I),b(J);
 c(I,J)=1; a(I)=1; b(J)=1;

POSITIVE VARIABLE x(I,J);
VARIABLE z;

EQUATIONS
 Cost(J)  funcion objetivo
 Rest(J)  restriccion;


Cost(J)$DIN(J)..    SUM(I,c(I,J)*x(I,J)) =e= z;
Rest(J)$DIN(J)..    SUM(I,a(I)*x(I,J))   =g=  b(J);

Model descompone /all/;

DIN(J)=NO;
loop(J1,
     DIN(J1)=YES;
     Solve descompone using lp minimizing z;
     DIN(J1)=NO;
);
```

13.3.5 Adición iterativa de restricciones a un problema

Este truco es de utilidad para algoritmos que resuelven un problema de forma iterativa. La solución óptima del problema de partida se consigue iterativamente mediante la resolución de problemas intermedios a los que se van añadiendo restricciones. Las restricciones que se añaden tienen la misma estructura pero difieren en los datos. Para programar este tipo de algoritmos es necesario el uso de bucles y conjuntos dinámicos. En primer lugar, se inicia el conjunto dinámico al conjunto vacío. Luego, este conjunto se actualiza añadiendo el índice que referencia la nueva restricción en cada iteración. Considérese la siguiente secuencia de problemas.

Minimizar

$$Z = cx$$

sujeto a

$$ax \geq b^{(i)}, \quad i = 1, \ldots, t$$

Supóngase que el número máximo de iteraciones que necesita el algoritmo es 5, ($t = 5$), y también que el escalar $b^{(i)}$ se incrementa en uno tras cada iteración; entonces se puede resolver el problema anterior en 5 iteraciones utilizando el siguiente código:

```
SETS   I contador de iteraciones  /1*5/
       DIN(I) conjunto dinamico;

ALIAS(I,I1);
```

```
SCALARS c,a;

PARAMETER b(I);
 c=1; a=1; b(I)=1;

POSITIVE VARIABLE x;
VARIABLE z;

EQUATIONS
 Cost     funcion objetivo
 Rest(I)  restriccion;

Cost..              c*x =e= z;
Rest(I)$DIN(I)..    a*x =g= b(I);

Model adicion /all/;

DIN(I)=NO;
loop(I1,
     DIN(I1)=YES;
     Solve adicion using lp minimizing z;
     b(I+1)=b(I)+1;
);
```

Para comprender mejor el funcionamiento del código anterior, el lector debe estudiar el fichero de salida GAMS.

13.3.6 Tratamiento de los estados inicial y final

En muchos problemas de ingeniería es importante establecer los estados inicial y final de ciertas variables. En GAMS, estos estados se pueden tratar mediante condiciones sobre las variables y restricciones que los modelan. Como siempre, el programador debe definir un conjunto ordenado para recorrer las variables en cuestión. En ese conjunto ordenado se puede distinguir mediante distintos identificadores un índice para el estado inicial y otro para el estado final. En GAMS, siempre es posible referirse al primer y último elemento de un conjunto que representarán respectivamente, el estado inicial y final. Por ejemplo, dado el conjunto K de períodos de tiempo, las condiciones para referirse a distintos subconjuntos son:

- `$(ord(K) EQ 1)` para el instante inicial
- `$(ord(K) EQ card(K))` para el instante final
- `$((ord(K) GT 1) AND (ord(K) LT card(K)))` para todos los instantes, excepto el inicial y el final.

Para más detalle, véanse los ejemplos de planificación de la producción y de la programación horaria de centrales térmicas en los apartados 11.2.2 y 11.3.6, respectivamente.

13.3.7 Análisis de sensibilidad

En ciertas aplicaciones, puede ser de interés establecer la dependencia de la solución respecto a algunos parámetros. Para ello, el problema se resuelve repetidas veces cambiando algunos datos de entrada o límites de sus variables.

A continuación se muestra, a modo de ejemplo, la dependencia de la solución de un problema respecto a un coeficiente de una restricción. El valor del coeficiente se modifica en cada iteración. Minimizar

$$Z = cx$$

sujeto a

$$a^{(t)}x \geq b.$$

El coeficiente a se incrementa en una unidad en cada iteración, es decir, $a^{(t+1)} = a^{(t)} + 1$. El siguiente código resuelve el problema 5 veces variando el valor de a.

```
SET   I  contador de iteraciones /1*5/

SCALARS a,b,c;
 a=1; b=1; c=1;

POSITIVE VARIABLE x;

VARIABLE z;

EQUATIONS
 Cost      funcion objetivo
 Rest      restriccion;

Cost..    c*x =e= z;
Rest..    a*x =g= b;

Model analisis /all/;

loop(I,
     Solve analisis using lp minimizing z;
     a=a+1;
);
```

13.3.8 Dependencia del flujo del programa

En ciertos casos, es interesante variar la estructura de un modelo según la información que se obtiene de problemas resueltos con anterioridad. Por ejemplo, si tras la resolución de un problema se concluye que éste no está acotado, se puede acotar añadiendo una nueva restricción al problema anterior y resolviéndolo de nuevo. El siguiente problema muestra un ejemplo sencillo. Minimizar

$$Z = cx$$

Si la solución del problema anterior está acotada se termina la ejecución del programa; si no, se añade un nueva restricción, de forma que el problema anterior se reescribe como sigue. Minimizar

$$Z = cx$$

sujeto a

$$ax \geq b$$

El siguiente código refleja la resolución condicionada de este último problema. En el código se incluyen algunos comentarios aclaratorios.

```
SCALARS c,a,b,cond;
 c=1; a=1; b=1;

VARIABLE x,z;

EQUATIONS
 COST      funcion objetivo
 REST      restriccion adicional;

COST..                c*x =e= z;
REST$(cond eq 1)..    a*x =g= b;

MODEL depende /ALL/;

** Si cond es 0 no se incluye en el modelo la restriccion REST
   cond=0;

** Se resuelve el modelo sin la restriccion REST
   SOLVE depende USING lp MINIMIZING z;

** Si el problema no esta acotado se agrega la restriccion REST
** y se vuelve a resolver

   if(depende.modelstat eq 3,

**   Se incluye en el modelo la restriccion REST
     cond=1;

**   El modelo se resuelve con la nueva restriccion
     SOLVE depende USING lp MINIMIZING z;
   );
```

Para comprender mejor el funcionamiento del programa, el lector debe estudiar el fichero de salida GAMS

Ejercicios

13.1 Un problema de estimación se puede formular como

$$\text{Minimizar}_{\Theta \in C} \;\; F(\hat{\mathbf{y}}, \mathbf{y}(\Theta))$$

donde $\hat{\mathbf{y}}$ son los datos observados, e $\mathbf{y}(\Theta)$ son los valores predichos por un modelo en función del vector de parámetros que se quiere estimar. F puede ser cualquier relación entre los datos observados y los datos predichos. Si F es la relación que viene dada por la norma euclídea ($\| \cdot \|_2$), el método de estimación se denomina método de los mínimos cuadrados (MMC), y F será igual a $\sum_{i=1}^{n} (\hat{y}_i - y_i)^2$. Si F viene dada por la norma ℓ_1 ($\| \cdot \|_1$), es decir, $F(\hat{\mathbf{y}}, \mathbf{y}) = \sum_{i=1}^{n} |\hat{y}_i - y_i|$, el método se denomina el *método del mínimo valor absoluto* (MVA). Otra posibilidad es expresar F como la norma del supremo ($\| \cdot \|_\infty$) lo cual implica que $F(\hat{\mathbf{y}}, \mathbf{y}) = \max \{|\hat{y}_i - y_i| \; i = 1, \cdots, n\}$. Este método se denomina *método de estimación min-max* (MM). Considérense los dos modelos teóricos que describen la variable y_i como una función de los parámetros a y b

$$
\begin{array}{lrcl}
\text{Modelo lineal} & y_i & = & a + bx_i \\
\text{Modelo no lineal} & y_i & = & (x_i - a)^b
\end{array}
$$

Estimar los parámetros a y b suponiendo que $(a, b) \in \mathbb{R}^2$ en ambos modelos, según los datos de entrada de la Tabla 13.1, mediante los métodos MMC, MVA y MM.

Tabla 13.1: Datos de entrada para el problema de estimación

x_i	$\hat{y}_i$
0	1.4
1	3.8
1.5	6.7
2	9.2
3	15.2
4	24.3
5	30.2

13.2 Considérese la siguiente secuencia de problemas. Minimizar

$$Z = x$$

sujeto a

$$a^{(i)} x \geq b^{(i)} \quad i = 1, \ldots, t$$

Supóngase que ($t = 5$), y también que los parámetros ($a^{(i)}, b^{(i)}$) vienen dados por la Tabla 13.2. Escribir un código en GAMS para resolver la secuencia de problemas anteriores.

Tabla 13.2: Parámetros para la secuencia de problemas

$a^{(i)}$	$b^{(i)}$
1	1.7
2	2.4
3	3.8
4	4.2
5	5.7

13.3 Sea un circuito eléctrico con 3 nudos y 3 líneas. El límite de capacidad y la susceptancia de las líneas (véase la sección 1.8 del capítulo 1) vienen dados por la siguiente tabla:

Línea	Susceptancia	Límite
1–2	2.5	0.3
1–3	3.5	0.7
2–3	3.0	0.7

La demanda eléctrica en el nudo 3 es de 0.85 MW. Existen dos generadores ubicados en los nudos 1 y 2, respectivamente. Los límites de producción de estos generadores son los siguientes:

Generador	Límite superior	Límite inferior
1	0.9	0
2	0.9	0

El coste de producción de los generadores se expresa como

$$c_i = \begin{cases} 0 & \text{if } p_i = 0 \\ F_i + V_i p_i & \text{if } p_i > 0 \end{cases}$$

donde F_i es el coste fijo, V_i es el coste variable, y p_i es la potencia de salida.

Los costes fijo y variable de los dos generadores viene dados por la siguiente tabla:

Generador	F_i	V_i
1	10	6
2	5	7

Determínese la producción de cada generador de forma que el coste total al suministrar la demanda sea mínimo. Repítase el problema tomando $F_2 = 10$.

13.4 Sea el siguiente problema de programación lineal. Minimizar

$$Z = x_1$$

sujeto a

$$\begin{array}{rrcr} -2x_1 & +\, x_2 & \leq & 0 \\ x_1 & -\, x_2 & \leq & 2 \\ x_1 & +\, x_2 & \leq & 6 \end{array}$$

(a) Resuélvase gráficamente

(b) Exprésese el problema en forma estándar considerando las siguientes alternativas:

  (i) Tratamiento de las variables no acotadas como la diferencia entre dos variables no negativas

  (ii) Tratamiento de las variables no acotadas añadiendo una sola variable

(c) Empléese GAMS para resolver el problema formulado con las dos alternativas y comparar los resultados

13.5 Sea el siguiente problema de programación lineal fraccional. Minimizar

$$Z = \frac{-2x_1 - 2x_3 + 3}{x_1 + x_2 + x_3 + 1}$$

sujeto a

$$\begin{array}{rrcr} -2x_1 & +\, x_2 & \leq & 0 \\ & -\, x_2 & \leq & 0 \\ x_1 & -\, x_2 & \leq & 2 \\ x_1 & +\, x_2 & \leq & 6 \end{array}$$

(a) Empléese GAMS para resolver el problema de programación lineal fraccional

(b) Formúlese este problema como un problema de programación lineal

(c) Empléese GAMS para resolver el problema de programación lineal y comparar los resultados con los obtenidos en (a).

13.6 Sean los dos conjuntos de restricciones

$$A = \left\{ \begin{array}{rrcr} -2x_1 & +\, x_2 & \leq & 0 \\ x_1 & -\, x_2 & \leq & 2 \\ x_1 & +\, x_2 & \leq & 6 \end{array} \right.$$

$$B = \left\{ \begin{array}{rrcr} -x_1 & +\, x_2 & \leq & -6 \\ -x_1 & +\, x_2 & \leq & -2 \\ 3x_1 & -\, x_2 & \leq & 18 \end{array} \right.$$

Tabla 13.3: Distribución de la temperatura dentro de la pared

Distancia	0	0.2	0.4	0.6	0.8	1.0
Temperatura	400	350	250	175	100	50

(a) Exprésese matemáticamente el siguiente problema. Minimizar

$$Z = x_1 - 3x_2$$

sujeto a

$$(x_1, x_2) \quad \in \quad A \cup B$$

(b) Resuélvase gráficamente este problema

(c) Empléese GAMS para resolver este problema. Modifíquese el código GAMS para obtener una solución óptima alternativa

13.7 Sea un problema de programación lineal con restricciones de igualdad y desigualdad. Escríbase un problema equivalente con sólo restricciones de desigualdad.

13.8 Sea un problema de programación matemática con una función objetivo no lineal. Escríbase un problema equivalente con una función objetivo lineal.

13.9 La Tabla 13.3 muestra la variación de temperatura dentro de una pared, que se calienta por una de sus caras, respecto a la distancia de la zona en la que se aplica el calor. Se quiere ajustar estos datos a un modelo lineal que exprese la temperatura como una función de la distancia

$$T = a + bx$$

donde a y b son constantes. Si los parámetros se estiman minimizando los errores siguientes, determínense estos coeficientes, por medio de GAMS y los trucos que se describen en este capítulo.

(a)

$$\sum_i \|a + bx_i - T_i\|$$

(b)

$$\max_i \|a + bx_i - T_i\|$$

donde las parejas (x_i, T_i) vienen dadas por la Tabla 13.3.

13.10 Dada una matriz $T(i, j)$, escribir una sola sentencia de GAMS que permita contar los elementos de la diagonal principal que son distintos de 10.

13.11 En GAMS, una vez se fija el valor de una variable, ésta permanece constante a lo largo del tiempo de ejecución. Supóngase que se requiere resolver un problema mediante un algoritmo iterativo en que el valor de la variable es fijo, pero distinto en cada iteración. Escríbase el código GAMS para abordar este caso.

Apéndice A

Conjunto de todas las soluciones factibles y compatibilidad

En el Capítulo 5 se ha visto que el conjunto de todas las soluciones factibles de un sistema lineal de ecuaciones e inecuaciones es un poliedro, que puede degenerar en un conjunto vacío, un espacio vectorial, un cono o un politopo o una suma de estas tres estructuras. El objeto de este apéndice consiste en estudiar el conjunto de todas las soluciones factibles y resolver los tres problemas siguientes:

1. Determinar si existe una solución factible, es decir, si el conjunto factible no es vacío

2. Obtener el conjunto de todas las soluciones, es decir, identificar todos los elementos de este conjunto

3. Interpretar el conjunto de las soluciones factibles de muchos de los ejemplos prácticos introducidos en los capítulos precedentes

La resolución de estos problemas requiere herramientas adecuadas que no son fáciles de encontrar en la literatura existente. Por ello, se han incluido en este apéndice los conceptos y las herramientas necesarias para trabajar con sistemas de ecuaciones e inecuaciones lineales.

El apéndice se ha estructurado como sigue. En la Sección A.1 se introduce el concepto de cono dual, que juega un importante papel en la discusión de la compatibilidad y en la resolución de sistemas de restricciones lineales. En la Sección A.2 se muestra cómo se puede obtener el cono asociado a un poliedro. En la Sección A.3 se da el algoritmo Γ que permitirá obtener el cono dual y encontrar el conjunto de todas las soluciones factibles. En la Sección A.4 se trata el problema de la compatibilidad de un conjunto de restricciones. En la Sección

A.5 se obtiene el conjunto de todas las soluciones factibles. Finalmente, en la Sección A.6 se discute el problema de la compatibilidad y se busca el conjunto de todas las soluciones factibles de algunos de los problemas discutidos en los capítulos precedentes, dando además la interpretación física de los resultados.

Como se ha mencionado anteriormente, aunque las restricciones no lineales son importantes, este capítulo se dedica sólo al caso de restricciones lineales.

Se han omitido la mayor parte de las demostraciones de este capítulo por considerar que se trata de resultados clásicos. El lector interesado en éstos puede consultar textos clásicos de análisis convexo, como, por ejemplo, Rockafellar [95]. En otras palabras, sólo se dan demostraciones detalladas del material relevante del capítulo.

A.1 El cono dual

Definición A.1 (dual no positivo o cono polar). *Sea* $\mathbf{A}_\pi$ *un cono en* $\mathbb{R}^n$ *con generadores* $\mathbf{a}_1, \ldots, \mathbf{a}_k$. *El dual no positivo* $\mathbf{A}_\pi^*$ *de* $\mathbf{A}_\pi$ *se define como el conjunto*

$$\mathbf{A}_\pi^* \equiv \left\{\mathbf{u} \in \mathbb{R}^n \mid \mathbf{A}^T\mathbf{u} \leq \mathbf{0}\right\} \equiv \left\{\mathbf{u} \in \mathbb{R}^n \mid \mathbf{a}_i^T\mathbf{u} \leq 0;\ i = 1, \ldots, k\right\}$$

es decir, el conjunto de todos los vectores tales que sus productos escalares por todos los vectores de $\mathbf{A}_\pi$ *son no positivos.* ■

Usando el teorema 5.4 se puede mostrar que $\mathbf{A}_\pi^*$ es también un cono, que se conoce como el *cono dual.* Utilizando la forma general de un cono (ver la Definición 5.9), se puede escribir el cono dual $\mathbf{A}_\pi^*$ de $\mathbf{A}_\pi$ en la forma $\mathbf{A}_\pi^* \equiv \mathbf{V}_\rho + \mathbf{W}_\pi$, donde $\mathbf{V}$ es el conjunto de los generadores de un espacio vectorial y $\mathbf{W}$ es el conjunto de generadores de un cono propio.

Nota A.1 *El dual no positivo de un subespacio vectorial coincide con su subespacio ortogonal:*

$$\begin{aligned} \mathbf{A}_\rho^* &\equiv (\mathbf{A}, -\mathbf{A})_\pi^* \equiv \{\mathbf{u} \in \mathbb{R}^n | \mathbf{a}_i^T\mathbf{u} \leq 0 \ \textit{and} \ -\mathbf{a}_i^T\mathbf{u} \leq 0; i = 1, \ldots, k\} \\ &\equiv \{\mathbf{u} \in \mathbb{R}^n | \mathbf{a}_i^T\mathbf{u} = 0; i = 1, \ldots, k\} \end{aligned}$$

■

Nota A.2 *El cono* $\mathbf{Ax} \leq \mathbf{0}$, *es decir, el conjunto factible del sistema de inecuaciones lineales* $\mathbf{Ax} \leq \mathbf{0}$, *coincide con el dual del cono generado por las filas de* $\mathbf{A}$. *Por tanto, obtener los generadores de* $\mathbf{Ax} \leq \mathbf{0}$ *es lo mismo que obtener los generadores del cono dual del cono generado por las filas de* $\mathbf{A}$. *Por ello, resolver el sistema de ecuaciones e inecuaciones* $\mathbf{Ax} \leq \mathbf{0}$ *es equivalente a obtener los generadores del cono dual del cono generado por las filas de* $\mathbf{A}$. ■

Nótese que esta nota es consecuencia inmediata de la definición de cono dual.

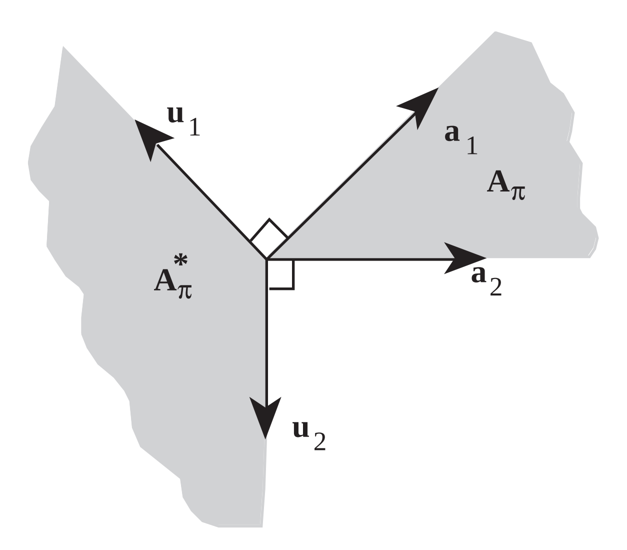


Figura A.1: Ejemplo de un cono $\mathbf{A}_\pi$ y su cono dual, $\mathbf{A}_\pi^*$.

Ejemplo A.1 (cono dual). La Figura A.1 muestra el cono dual, $\mathbf{A}_\pi^* \equiv (\mathbf{u}_1, \mathbf{u}_2)_\pi$, del cono $(\mathbf{a}_1, \mathbf{a}_2)_\pi$.

■

El cono dual satisface las propiedades siguientes:

1. El dual es único: $\mathbf{A}_\pi \equiv \mathbf{B}_\pi \iff \mathbf{A}_\pi^* \equiv \mathbf{B}_\pi^*$.
2. El dual de un subconjunto contiene al dual de cualquier superconjunto, es decir
$$\mathbf{E}_\pi \subseteq \mathbf{A}_\pi \iff \mathbf{A}_\pi^* \subseteq \mathbf{E}_\pi^*.$$

El lema siguiente se debe a Farkas (1902) y es la base de la teoría de dualidad.

Lema A.1 (Farkas). *Si $\mathbf{A}_\pi^*$ es el dual no positivo de $\mathbf{A}_\pi$, entonces*

$$\mathbf{x} \in \mathbf{A}_\pi \Leftrightarrow \boldsymbol{\mu}^T\mathbf{x} \leq 0; \forall \boldsymbol{\mu} \in \mathbf{A}_\pi^*. \tag{A.1}$$

En otras palabras, el dual del dual de $\mathbf{A}_\pi$ es $\mathbf{A}_\pi$ mismo: $\mathbf{A}_\pi^{**} \equiv \mathbf{A}_\pi$.

Este lema conduce al teorema siguiente, que permite determinar si un vector pertenece a un cono.

Teorema A.1 (vector en un cono). *El vector $\mathbf{x}$ pertenece al cono $\mathbf{A}_\pi$ si y sólo si $\mathbf{V}^T\mathbf{x} = \mathbf{0}$ y $\mathbf{W}^T\mathbf{x} \leq \mathbf{0}$:*

$$\mathbf{x} \in \mathbf{A}_\pi \Leftrightarrow \mathbf{V}^T\mathbf{x} = \mathbf{0} \text{ y } \mathbf{W}^T\mathbf{x} \leq \mathbf{0}, \tag{A.2}$$

donde $\mathbf{V}_\rho + \mathbf{W}_\pi$ es la forma general del cono dual $\mathbf{A}_\pi^*$ ■

Demostración. El cono dual $\mathbf{A}_\pi^*$ puede escribirse como

$$\mathbf{A}_\pi^* \equiv \mathbf{U}_\pi \equiv \mathbf{V}_\rho + \mathbf{W}_\pi \equiv (\mathbf{V} : -\mathbf{V} : \mathbf{W})_\pi \tag{A.3}$$

Según el lema de Farkas, se tiene

$$\begin{array}{rcl} \mathbf{x} \in \mathbf{A}_\pi & \Leftrightarrow & \mathbf{U}^T\mathbf{x} \leq \mathbf{0} \Leftrightarrow \mathbf{V}^T\mathbf{x} \leq \mathbf{0} \text{ y } -\mathbf{V}^T\mathbf{x} \leq \mathbf{0} \text{ y } \mathbf{W}^T\mathbf{x} \leq \mathbf{0} \\ & \Leftrightarrow & \mathbf{V}^T\mathbf{x} = \mathbf{0} \text{ y } \mathbf{W}^T\mathbf{x} \leq \mathbf{0} \end{array} \tag{A.4}$$

■

A.2 Cono asociado a un poliedro

Los conos duales son muy útiles para obtener conjuntos de generadores minimales del espacio vectorial, el cono y el politopo que definen el poliedro de las soluciones factibles. Con este objetivo, se reemplaza nuestro espacio euclídeo $\mathbb{R}^n$por otro espacio euclídeo $\mathbb{R}^{n+1}$ (ver Castillo et al. [21]). Usando este truco, se convierten los poliedros en $\mathbb{R}^n$ en conos en $\mathbb{R}^{n+1}$. Por ello, sólo tiene que tratarse con conos. Sin embargo, al final, se necesita volver al espacio euclídeo de partida $\mathbb{R}^n$.

Más precisamente, el poliedro

$$S = \{\mathbf{x} \in \mathbb{R}^n | \mathbf{H}\mathbf{x} \leq \mathbf{a}\}$$

se reemplaza por el cono

$$\mathbf{C}_S = \left\{ \begin{pmatrix} \mathbf{x} \\ x_{n+1} \end{pmatrix} \in \mathbb{R}^{n+1} | \mathbf{H}\mathbf{x} - x_{n+1}\mathbf{a} \leq \mathbf{0};\ -x_{n+1} \leq 0 \right\}$$

Este proceso se ilustra en la Figura A.2, donde se comienza con el pentágono (poliedro) en $\mathbb{R}^2$ y se asciende a $\mathbb{R}^3$ añadiendo una componente más a sus vértices (puntos extremos). Seguidamente, se considera el cono generado por estas direcciones extremas. Nótese que los puntos extremos en $\mathbb{R}^2$ se transforman en direcciones extremas en $\mathbb{R}^3$. Una vez que estamos en $\mathbb{R}^3$, se buscan los generadores del cono dual, usando el algoritmo Γ que se describe más adelante. Finalmente, se recuperan las restricciones iniciales en $\mathbb{R}^2$ imponiendo la restricción $x_n = 1$, es decir, obteniendo la intersección del cono con el plano (hiperplano) $x_n = 1$.

Es importante hacer notar que si el poliedro tiene direcciones, éstas generarán direcciones en el cono asociado. El teorema que sigue relaciona los generadores del poliedro y de su cono asociado.

Teorema A.2 (generadores del poliedro y de su cono asociado). *Sea* $\mathbf{H}$ *una matriz y considérese el poliedro* $\mathbf{H}\mathbf{x} \leq \mathbf{a}$. *Entonces, el sistema de inecuaciones* $\mathbf{H}\mathbf{x} \leq \mathbf{a}$ *es equivalente a*

$$\begin{array}{rcl} \mathbf{C}\begin{pmatrix} \mathbf{x} \\ x_{n+1} \end{pmatrix} & \leq & \mathbf{0} \\ x_{n+1} & = & 1 \end{array}$$

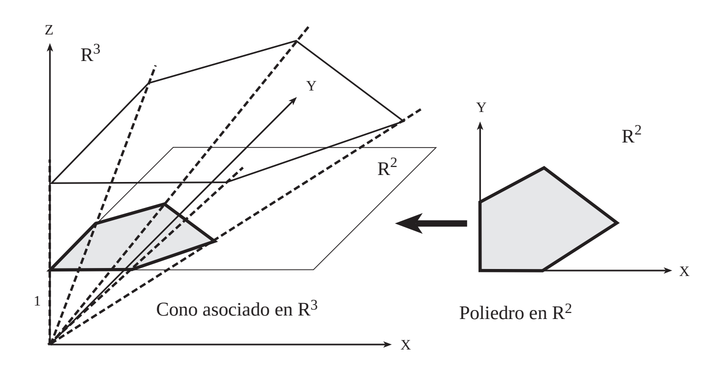


Figura A.2: Ilustración de cómo se obtiene el cono asociado a un poliedro.

donde

$$\mathbf{C} = \begin{pmatrix} \mathbf{H} & | & -\mathbf{a} \\ -- & + & -- \\ \mathbf{0} & | & -1 \end{pmatrix}$$

Si $\mathbf{C}_\pi^{T*} = \mathbf{V}_\rho^0 + \mathbf{W}_\pi^0$ *es la forma general del cono* $\mathbf{C}_\pi^{T*}$*, donde* $\mathbf{W}^0$ *tiene todas sus últimas componentes iguales a cero o uno, entonces, la forma general del poliedro* $\mathbf{Hx} \leq \mathbf{a}$ *es*

$$S = \mathbf{V}_\rho + \mathbf{W}_\pi + \mathbf{Q}_\lambda$$

donde $\mathbf{V}$ *es* $\mathbf{V}^0$ *con sus últimas componentes eliminadas, y* $\mathbf{W}$ *y* $\mathbf{Q}$ *contienen todos los vectores columna de* $\mathbf{W}^0$ *tales que sus últimas componentes son nulas o no nulas, respectivamente, con sus últimas componentes eliminadas.* ■

Demostración. Supóngase que $\mathbf{C}_\pi^{T*} = \mathbf{V}_\rho^0 + \mathbf{W}_\pi^0$. Entonces, todos los vectores en $\mathbf{V}^0$ deben tener su última componente nula, debido a la restricción redundante $-x_{n+1} \leq 0$. Además, y sin pérdida de generalidad, se puede suponer que w_{n+1k}^0, la última componente de cualquier vector en $\mathbf{W}^0$, es 0 ó 1, puesto que si no lo fuera, se podría dividir el vector por w_{n+1k}^0.

Finalmente, la condición $x_{n+1} = 1$ conduce a

$$\sum_{k \in K_{W^0}} \pi_k = 1 \tag{A.5}$$

donde

$$K_{W^0} \equiv \{k | w_{n+1k}^0 \neq 0\} \tag{A.6}$$

es el conjunto de todos los vectores de $\mathbf{W}^0$ que tienen no nula su última componente.

Se tienen los dos casos siguientes:

1. $K_{W^0} \equiv \emptyset$: En este caso el poliedro es vacío, ya que esto implica $x_{n+1} = 0 \neq 1$.

2. $K_{W^0} \neq \emptyset$: Haciendo $\lambda_k = \pi_k$, (A.5) se obtiene

$$\sum_{k \in K_{W^0}} \lambda_k = 1 \tag{A.7}$$

y, volviendo a nuestro espacio de partida $\mathbb{R}^n$, el cono $\mathbf{V}^0_\rho + \mathbf{W}^0_\pi$ se transforma en el poliedro $\mathbf{V}_\rho + \mathbf{W}_\pi + \mathbf{Q}_\lambda$, donde $\mathbf{W}$ y $\mathbf{Q}$ son matrices cuyas columnas tienen la última componente nula y unidad, respectivamente, y sus últimas componentes han sido eliminadas.

■

Por ello, si se tiene un procedimiento para obtener los generadores del cono, se tiene, inmediatamente, un procedimiento para obtener los generadores del poliedro. Uno de estos métodos ha sido desarrollado por Castillo et al. [21], y se conoce como el "algoritmo Γ", que se describirá en la Sección siguiente.

Ejemplo A.2 (generadores del poliedro y su cono asociado). Considérese el poliedro S:

$$\begin{array}{rrcl} -x_2 & & \leq & 0 \\ & -x_3 & \leq & 0 \\ & x_3 & \leq & 1 \end{array} \tag{A.8}$$

que se puede escribir como

$$\begin{array}{rrrcl} -x_2 & & & \leq & 0 \\ & -x_3 & & \leq & 0 \\ & x_3 & -x_4 & \leq & 0 \\ & & -x_4 & \leq & 0 \\ & & x_4 & = & 1 \end{array} \tag{A.9}$$

o como

$$\mathbf{C}\begin{pmatrix} \mathbf{x} \\ x_4 \end{pmatrix} = \begin{pmatrix} 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & -1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} \leq \mathbf{0}; \quad x_4 = 1 \tag{A.10}$$

El dual $\mathbf{C}^{T^*}_\pi$ es el conjunto de vectores de la forma

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \rho_1 \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} + \pi_1 \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} + \pi_2 \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \end{pmatrix} \Rightarrow \mathbf{V}^0 = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \quad \mathbf{W}^0 = \begin{pmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{pmatrix} \tag{A.11}$$

Nótese que todas las últimas componentes son la unidad o nulas.

Entonces, según el Teorema A.2, el poliedro es

$$S = \mathbf{V}_\rho + \mathbf{W}_\pi + \mathbf{Q}_\lambda = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}_\rho + \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}_\pi + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}_\lambda, \tag{A.12}$$

es decir, los vectores del poliedro son de la forma [véase la Figura 5.1(c)]

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \rho_1 \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} + \pi_1 \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} + \lambda_1 \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \tag{A.13}$$

■

A.3 El procedimiento $\boldsymbol{\Gamma}$

En esta sección se explica el algoritmo Γ, que permite resolver sistemas lineales de ecuaciones e inecuaciones, calculando, con un conjunto de generadores mínimo, el cono dual $\mathbf{A}_\pi^*$ de $\mathbf{A}_\pi$ en su forma general.

El proceso computacional asociado a lo que se ha llamado el algoritmo Γ ha sido tratado por Jubete [61, 62]. Este procedimiento permite obtener el dual $\mathbf{A}_\pi^* \equiv \mathbf{V}_\rho + \mathbf{W}_\pi$ de $\mathbf{A}_\pi \equiv \mathbf{B}_\rho + \mathbf{C}_\pi$:

$$\begin{array}{rcl} \mathbf{A}_\pi^* & \equiv & \{\boldsymbol{\mu} \in \mathbb{R}^n | \boldsymbol{\mu}^T \mathbf{a}_i \leq 0; i = 1, \ldots, k\} \\ & \equiv & \{\boldsymbol{\mu} \in \mathbb{R}^n | \boldsymbol{\mu}^T \mathbf{b}_i = 0, \boldsymbol{\mu}^T \mathbf{c}_j \leq 0, \forall i, j\} \end{array} \tag{A.14}$$

con un conjunto mínimo de generadores. Seguidamente se da una descripción detallada de este algoritmo. Más detalles y ejemplos pueden verse en Castillo et al. [21]. El lector no debe olvidar que el objetivo final es obtener la solución general de los sistemas de inecuaciones. Los duales son sólo una etapa intermedia.

Algoritmo A.1 (el Algoritmo $\boldsymbol{\Gamma}$).

- **Entrada.** Un cono en el espacio euclídeo $\mathbb{R}^n$, definido por un conjunto de generadores $\mathbf{A} = \{\mathbf{a}_1, \ldots, \mathbf{a}_m\}$.

- **Salida.** Los generadores del cono dual.

Iniciación.

- Sea $\mathbf{U} = \mathbf{I}_n$, donde $\mathbf{I}_n$ es la matriz identidad de dimensión n, y sea V la lista (conjunto ordenado) de índices de 1 a n.

- Iniciar una lista $A_0(\mathbf{u}_j)$ (la lista de vectores de $\mathbf{A}$ que son ortogonales a $\mathbf{u}_j$ (columna de $\mathbf{U}$)) para cada vector $\mathbf{u}_j; j = 1, 2, \ldots, n$ al conjunto vacío, y hacer $h = 1$.

Etapa 1. (calcular los productos escalares). Calcular $\mathbf{t} = \mathbf{a}_h^T \mathbf{U}$.

Etapa 2. (buscar el pivote). Encontrar la columna pivote ($t_{pivote} \neq 0$) entre las columnas con índices en V.

Etapa 3. (decidir entre los procesos 1 y 2). Si no se ha encontrado un pivote, ir al Proceso 2 (Etapa 5). En otro caso, ir al Proceso 1 (etapa 4).

Etapa 4. (proceso 1). Normalizar la columna pivote dividiéndola por $-t_{pivote}$ (nótese que el pivote es un índice), y realizando la operación de pivotación haciendo $u_{ij} = u_{ij} + t_j u_{i\,pivote}$ para todo $j \neq pivote$. Añadir el índice h a las listas $A_0(\mathbf{u}_j)$ tales que $j \neq pivote$. Eliminar el índice *pivote* de V y cambiar el vector $\mathbf{u}_{pivote}$ de $\mathbf{v}$ a $\mathbf{w}$. Seguidamente, ir a la etapa 6.

Etapa 5. (proceso 2). Obtener los conjuntos

$$\begin{array}{lcl} \mathbf{I}^- & \equiv & \{i|\mathbf{w}_i^T\mathbf{a}_h < 0; i \notin V\} \\ \mathbf{I}^+ & \equiv & \{i|\mathbf{w}_i^T\mathbf{a}_h > 0; i \notin V\} \\ \mathbf{I}^0 & \equiv & \{i|\mathbf{w}_i^T\mathbf{a}_h = 0; i \notin V\} \end{array} \tag{A.15}$$

y añadir a $A_0(\mathbf{u}_j)$ el índice h para todo $j \in \mathbf{I}_0$. Si $\mathbf{I}^+ \equiv \emptyset$, ir a la etapa 6. En otro caso, si $\mathbf{I}^- \equiv \emptyset$, eliminar de $\mathbf{U}$ todos los vectores $\mathbf{u}_j$; $j \in \mathbf{I}^+$. En otro caso, para cada $i \in \mathbf{I}^-$ y $j \in \mathbf{I}^+$ construir el conjunto de vectores

$$\mathbf{W}^\pm \equiv \{\mathbf{w}|\mathbf{w} = t_j\mathbf{w}_i - t_i\mathbf{w}_j\}$$

e iniciar para cada conjunto $\mathbf{w} \in \mathbf{W}^\pm$ un lista $\mathbf{A}_0(\mathbf{w})$ al conjunto $(A_0(\mathbf{u}_i) \cap A_0(\mathbf{u}_j)) \cup \{h\}$. Eliminar de $\mathbf{U}$ todos los vectores $\mathbf{u}_j; j \in \mathbf{I}^+$, y añadir a $\mathbf{U}$ los vectores $\mathbf{w} \in \mathbf{W}^\pm$ tales que

$$\mathbf{w} \neq \mathbf{0};\ A_0(\mathbf{w}) \not\subset A_0(\mathbf{w}^*), \mathbf{w} \neq \mathbf{w}^* \in \mathbf{W}^\pm;\ A_0(\mathbf{w}) \not\subset A_0(\mathbf{w}_j), j \in \mathbf{I}^0$$

Etapa 6. (proceso 2). Si $h < m$, hacer $h = h + 1$ e ir a la etapa 1; En otro caso, devolver la matriz $\mathbf{U}$, y salir.

■

Ejemplo A.3 (cono dual). Considérese el cono

$$\mathbf{A}_\pi = \begin{pmatrix} 1 & 2 & 0 & 0 & 1 & 0 \\ 0 & 1 & -1 & -1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 2 & 1 & 1 & -1 \end{pmatrix} \tag{A.16}$$

Iniciación. Se hace $\mathbf{U} = \mathbf{I}_n$, como se indica en la tabla A.1 (fase 1), y $V = \{1, 2, 3, 4\}$; es decir, en esta etapa inicial todos los vectores son de tipo $\mathbf{v}$, o lo que es igual, generadores del espacio vectorial. Se inician también 4 listas, una por cada vector $\mathbf{v}$ al conjunto vacío, y se hace $h = 1$.

Etapa 1. Se incorpora la primera columna de $\mathbf{A}$, $\mathbf{a}_1$, en la primera columna de la tabla A.1 (fase 1), se calculan los, productos escalares $\mathbf{t} = \mathbf{a}_1^T\mathbf{U}$, y se escriben en la última fila de la tabla A.1 (fase 1).

Tabla A.1: Vector incorporado, matriz **T** y productos escalares asociados

Fase 1					Fase 2				
$\mathbf{a}_1$	$\mathbf{v}_1^1$	$\mathbf{v}_2^1$	$\mathbf{v}_3^1$	$\mathbf{v}_4^1$	$\mathbf{a}_2$	$\mathbf{w}_1^2$	$\mathbf{v}_1^2$	$\mathbf{v}_2^2$	$\mathbf{v}_3^2$
1	**1**	0	0	0	2	–1	**0**	0	–1
0	**0**	1	0	0	1	0	**1**	0	0
0	**0**	0	1	0	0	0	**0**	1	0
1	**0**	0	0	1	0	0	**0**	0	1
$\mathbf{t}^1$	**1**	0	0	1	$\mathbf{t}^2$	–2	**1**	0	–2
		1	1		A_0		1	1	1
								2	

Tabla A.2: Vector incorporado, matriz **T** y productos escalares asociados

Fase 3					Fase 4				
$\mathbf{a}_3$	$\mathbf{w}_1^3$	$\mathbf{w}_2^3$	$\mathbf{v}_1^3$	$\mathbf{v}_2^3$	$\mathbf{a}_4$	$\mathbf{w}_1^4$	$\mathbf{w}_2^4$	$\mathbf{w}_3^4$	$\mathbf{v}_1^4$
0	–1	0	**0**	–1	0	–1	0	0	**–1**
–1	2	–1	**0**	2	–1	2	–1	0	**2**
1	0	0	**1**	0	1	2	–1	–1	**0**
2	0	0	**0**	1	1	0	0	0	**1**
$\mathbf{t}^3$	–2	1	**1**	0	$\mathbf{t}^4$	0	0	–1	**–1**
A_0	2	1	1	1	A_0	2	1	1	1
			2	2		3	3	2	2
				3		4	4		3

Etapa 2. Se busca el pivote, y se encuentra pivote=1.

Etapa 3. Puesto que se ha encontrado un pivote, se va al proceso 1.

Etapa 4. Se normaliza la columna pivote, se realiza la pivotación, se añade el índice 1 a las listas $A_0(\mathbf{u}_j)$ para todo $j \neq 1$, y se cambia el vector $\mathbf{u}_1$ de $\mathbf{v}$ a $\mathbf{w}$, obteniendo la matriz de la tabla A.1 (fase 2).

Etapa 6. Puesto que $h < 6$, se hace $h = 2$ y se va a la etapa 1.

Etapa 1. Se incorpora la columna segunda de $\mathbf{A}$, $\mathbf{a}_2$, en la primera columna de la tabla A.1 (fase 2), se calculan los productos escalares $\mathbf{t} = \mathbf{a}_2^T\mathbf{U}$, y se escriben en la última fila de la tabla A.1 (fase 2).

Etapa 2. Se busca el pivote, y se encuentra pivote=2.

Etapa 3. Puesto que se ha encontrado un pivote, se va al proceso 1.

Tabla A.3: Vector incorporado, matriz $\mathbf{T}$ y productos escalares asociados

Fase 5					Fase 6						
$\mathbf{a}_5$	$\mathbf{w}_1^5$	$\mathbf{w}_2^5$	$\mathbf{w}_3^5$	$\mathbf{w}_4^5$	$\mathbf{a}_6$	$\mathbf{w}_1^6$	$\mathbf{w}_2^6$	$\mathbf{w}_3^6$	$\mathbf{w}_4^6$	$\mathbf{w}_5^6$	$\mathbf{w}_6^6$
1	–1	0	1	–1	0	0	1	–2	–2	0	–1
1	2	–1	–2	2	0	–1	–2	1	2	0	2
1	2	–1	–1	0	0	–1	–1	1	–2	3	–2
1	0	0	–1	1	–1	0	–1	0	2	–3	1
$\mathbf{t}^5$	3	–2	–3	2	$\mathbf{t}^6$	0	1	0	–2	3	–1
A_0	2	1	1	1	A_0	1	1	3	1	2	1
	3	3	2	2		3	2	4	3	4	2
	4	4	4	3		4	4	5	5	5	5
						6		6			

Etapa 4. Se normaliza la columna pivote, se realiza la pivotación, se añade el índice 2 a las listas $A_0(\mathbf{u}_j)$ para todo $j \neq 2$, y se cambia el vector $\mathbf{u}_2$ de $\mathbf{v}$ a $\mathbf{w}$, obteniendo la matriz de la tabla A.2 (fase 3).

Etapa 6. Puesto que $h < 6$, se hace $h = 3$ y se va a la etapa 1.

Etapa 1. Se incorpora la tercera columna de $\mathbf{A}$, $\mathbf{a}_3$, en la primera columna de la tabla A.2 (fase 3), se calculan los productos escalares $\mathbf{t} = \mathbf{a}_3^T\mathbf{U}$, y se escriben en la última fila de la tabla A.2 (fase 3).

Etapa 2. Se busca el pivote, y se encuentra pivote=3.

Etapa 3. Puesto que se ha encontrado un pivote, se va al proceso 1.

Etapa 4. Se normaliza la columna pivote, se realiza la pivotación, se añade el índice 3 a las listas $A_0(\mathbf{u}_j)$ para todo $j \neq 3$, y se cambia el vector $\mathbf{u}_3$ de $\mathbf{v}$ a $\mathbf{w}$, obteniendo la matriz de la tabla A.2 (fase 4).

Etapa 6. Puesto que $h < 6$, se hace $h = 4$ y se va a la etapa 1.

Etapa 1. Se incorpora la cuarta columna de $\mathbf{A}$, $\mathbf{a}_4$, en la primera columna de la tabla A.2 (fase 4), se calculan los productos escalares $\mathbf{t} = \mathbf{a}_4^T\mathbf{U}$, y se escriben en la última fila de la tabla A.2 (fase 4).

Etapa 2. Se busca el pivote, y se encuentra pivote=4.

Etapa 3. Puesto que se ha encontrado un pivote, se va al proceso 1.

Etapa 4. Se normaliza la columna pivote, se realiza la pivotación, se añade el índice 4 a las listas $A_0(\mathbf{u}_j)$ para todo $j \neq 4$, y se cambia el vector $\mathbf{u}_4$ de $\mathbf{v}$ a $\mathbf{w}$, obteniendo la matriz de la tabla A.3 (fase 5).

Etapa 6. Puesto que $h < 6$, se hace $h = 5$ y se va a la etapa 1.

Etapa 1. Se incorpora la quinta columna de $\mathbf{A}$, $\mathbf{a}_5$, en la primera columna de la tabla A.3 (fase 5), se calculan los productos escalares $\mathbf{t} = \mathbf{a}_5^T\mathbf{U}$, y se escriben en la última fila de la tabla A.3 (fase 5).

Etapa 2. Se busca el pivote, pero no se encuentra (no existen vectores etiquetados $\mathbf{v}$).

Etapa 3. Puesto que no se ha encontrado pivote, se va al proceso 2.

Etapa 5. Se determinan los conjuntos

$$\mathbf{I}^- \equiv \{2,3\};\ \ \mathbf{I}^+ \equiv \{1,4\};\ \ \mathbf{I}^0 \equiv \{\}$$

Se construye el conjunto de vectores

$$\mathbf{W}^\pm \equiv \{(-2,1,1,0)^T, (-2,2,-2,2)^T, (0,0,3,-3)^T, (-1,2,-2,1)^T\}$$

y se inician las listas A_0 de éstos como $(A_0(\mathbf{u}_i) \cap A_0(\mathbf{u}_j)) \cup \{h\}$ (véanse los conjuntos A_0 en la tabla A.3 (fase 6)). Se eliminan de $\mathbf{U}$ todos los vectores $\mathbf{u}_j; j \in \mathbf{I}^+$, y, puesto que ninguno de los conjuntos $A_0(\mathbf{w})$ están contenidos en otros $A_0(\mathbf{w}^*)$ para $\mathbf{w}, \mathbf{w}^* \in \mathbf{W}^\pm$, y, además, no hay conjuntos $A_0(\mathbf{w}_j; j \in \mathbf{I}^0)$, se añaden los cuatro vectores de $\mathbf{W}^\pm$ a $\mathbf{U}$.

Etapa 6. Puesto que $h < 6$, se hace $h = 6$ y se va a la etapa 1.

Etapa 1. Se incorpora la sexta columna de $\mathbf{A}$, $\mathbf{a}_6$, en la primera columna de la tabla A.4 (fase 6), se calculan los productos escalares $\mathbf{t} = \mathbf{a}_6^T\mathbf{U}$, y se escriben en la última fila de la tabla A.4 (fase 6).

Etapa 2. Se busca el pivote y no se encuentra (no existen vectores etiquetados $\mathbf{v}$).

Etapa 3. Puesto que no se ha encontrado pivote, se va al proceso 2.

Etapa 5. Se determinan los conjuntos

$$\mathbf{I}^- \equiv \{4,6\};\ \ \mathbf{I}^+ \equiv \{2,5\};\ \ \mathbf{I}^0 \equiv \{1,3\}$$

Se construye el conjunto de vectores

$$\begin{array}{rcl}\mathbf{W}^\pm & \equiv & \{(-2,1,1,0)^T, (-2,2,-2,2)^T, (0,0,3,-3)^T \\ & & (-1,2,-2,1)^T, (0,-2,-4,0)^T, (-6,6,0,0)^T\}\end{array}$$

Tabla A.4: Dual sin normalizar

	$\mathbf{w}_1$	$\mathbf{w}_2$	$\mathbf{w}_3$	$\mathbf{w}_4$	$\mathbf{w}_5$	$\mathbf{w}_6$	$\mathbf{w}_7$	$\mathbf{w}_8$
A_0	0	–2	–2	–1	0	–3	0	–6
	–1	1	2	2	0	6	–2	6
	–1	1	–2	–2	–3	–3	–4	0
	0	0	2	1	0	0	0	0
A_0	1	3	1	1	1	6	1	5
	3	4	3	2	2		6	6
	4	5	5	5	6			
	6	6						

Tabla A.5: Dual normalizado

$\mathbf{w}_1$	$\mathbf{w}_2$	$\mathbf{w}_3$	$\mathbf{w}_4$	$\mathbf{w}_5$	$\mathbf{w}_6$
0	–2	–1	–1	0	–1
–1	1	1	2	0	2
–1	1	–1	–2	–1	–1
0	0	1	1	0	0

y se inician las listas A_0 de éstos como $(A_0(\mathbf{u}_i) \cap A_0(\mathbf{u}_j)) \cup \{h\}$ (véanse los conjuntos A_0 en la tabla A.4). Puesto que $A_0(\mathbf{w}_7) \subset A_0(\mathbf{w}_6)$ y $A_0(\mathbf{w}_8) \subset A_0(\mathbf{w}_6)$, sólo se añaden a $\mathbf{U}$ los vectores $\mathbf{w}_5$ y $\mathbf{w}_6$. Finalmente, si se normalizan todos los vectores, es decir, si se dividen los vectores con su última componente no nula por ella, se obtiene el dual normalizado de la tabla A.5.

Por ello, el cono dual del cono $\mathbf{A}_\pi$ en (A.16) es

$$\pi_1 \begin{pmatrix} 0 \\ -1 \\ -1 \\ 0 \end{pmatrix} + \pi_2 \begin{pmatrix} -2 \\ 1 \\ 1 \\ 0 \end{pmatrix} + \pi_3 \begin{pmatrix} -1 \\ 1 \\ -1 \\ 1 \end{pmatrix} + \pi_4 \begin{pmatrix} -1 \\ 2 \\ -2 \\ 1 \end{pmatrix} + \pi_5 \begin{pmatrix} 0 \\ 0 \\ -1 \\ 0 \end{pmatrix} + \pi_6 \begin{pmatrix} -1 \\ 2 \\ -1 \\ 0 \end{pmatrix}$$

∎

Una vez que ya se sabe cómo se puede obtener el cono dual $\mathbf{A}_\pi^*$ de un cono dado $\mathbf{A}_\pi$, se puede retornar a los dos problemas iniciales descritos al comienzo de este apéndice:

1. Determinar si existe alguna solución del sistema de inecuaciones $\mathbf{Hx} \le \mathbf{a}$, es decir, si el conjunto factible no es vacío

2. Obtener el conjunto de todas las soluciones factibles de $\mathbf{Hx} \le \mathbf{a}$, es decir, identificar todos los elementos de este conjunto.

Con este fin, se utiliza la transformación de $\mathbb{R}^n$ a $\mathbb{R}^{n+1}$ descrita al comienzo de esta sección A.3 y que se ilustra en la Figura A.2 para transformar el sistema $\mathbf{Hx} \le \mathbf{a}$ en el sistema

$$\mathbf{Hx} - \mathbf{a}x_{n+1} = \mathbf{H}^*\mathbf{x}^* \le \mathbf{0}; \;\; x_{n+1} = 1$$

donde $\mathbf{H}^*$ y $\mathbf{x}^*$ son las nuevas matrices de coeficientes y de incógnitas.

Seguidamente, se resuelve el primer problema en la sección siguiente utilizando el algoritmo Γ con el cono generado por las columnas de $\mathbf{H}^*$, y el segundo problema, usando el algoritmo Γ con el cono generado por las filas de $\mathbf{H}^*$.

A.4 Compatibilidad de sistemas lineales

En esta sección se analiza si el conjunto de soluciones factibles del sistema $\mathbf{Hx} \le \mathbf{a}$ es vacío.

Para analizar el problema de la compatibilidad, se necesita transformar el conjunto de restricciones en un conjunto de inecuaciones en variables no negativas. Esto no es un problema puesto que siempre se puede encontrar esta transformación. Por ejemplo

- En el caso de inecuaciones, se utilizan variables de holgura para reducir las inecuaciones a ecuaciones (se añade una nueva variable por cada inecuación).
- En el caso mixto de ecuaciones e inecuaciones, se reducen las inecuaciones como en el caso previo.
- En el caso mixto de variables no restringidas y restringidas, se reemplazan las variables no restringidas por diferencias de variables no negativas con una variable común en el sustraendo (se añade una sola variable más).

Por ello, se considera el sistema de ecuaciones lineales

$$\begin{aligned} \mathbf{Cx}_\pi &= \mathbf{a} \\ \mathbf{x}_\pi &\ge \mathbf{0} \end{aligned} \tag{A.17}$$

donde $\mathbf{C}$ es la matriz de coeficientes, y el subíndice π se utiliza para referirse a variables no negativas.

Puesto que el sistema (A.17) puede ser escrito en la forma

$$x_1 \begin{pmatrix} c_{11} \\ c_{21} \\ \vdots \\ c_{n1} \end{pmatrix} + x_2 \begin{pmatrix} c_{12} \\ c_{22} \\ \vdots \\ c_{n2} \end{pmatrix} + \ldots + x_n \begin{pmatrix} c_{1n} \\ c_{2n} \\ \vdots \\ c_{nn} \end{pmatrix} = \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{pmatrix}; \;\; x_i \ge 0, \; \forall i \tag{A.18}$$

tendrá solución si y sólo si el vector $\mathbf{a}$ es una combinación lineal no negativa de las columnas de $\mathbf{C}$, es decir, si $\mathbf{a}$ pertenece al cono generado por las columnas de $\mathbf{C}$ ($\mathbf{a} \in \mathbf{C}_\pi$). Usando el Teorema A.1, se obtiene

$$\mathbf{Cx}_\pi = \mathbf{a} \text{ tiene una solución } \Leftrightarrow \mathbf{V}^T\mathbf{a} = \mathbf{0} \text{ y } \mathbf{W}^T\mathbf{a} \le \mathbf{0} \tag{A.19}$$

donde $\mathbf{C}_\pi^* \equiv \mathbf{U}_\pi \equiv \mathbf{V}_\rho + \mathbf{W}_\pi$. Por ello, la comprobación de la compatibilidad se reduce a obtener el cono dual $\mathbf{C}_\pi^* \equiv \mathbf{U}_\pi \equiv \mathbf{V}_\rho + \mathbf{W}_\pi$ y comprobar (A.19).

Ejemplo A.4 (compatibilidad). Considérese el conjunto de restricciones

$$\begin{array}{rrrcl} x_1 & & & \leq & p \\ 2x_1 & +x_2 & & \leq & q \\ & -x_2 & +x_3 & \leq & r \\ x_1, & x_2, & x_3 & \geq & 0 \end{array} \tag{A.20}$$

que, usando las variables de holgura x_4, x_5 y x_6, pueden ser escritas como

$$\begin{array}{rrrrrrcl} x_1 & & & +x_4 & & & = & p \\ 2x_1 & +x_2 & & & +x_5 & & = & q \\ & -x_2 & +x_3 & & & +x_6 & = & r \\ x_1, & x_2, & x_3, & x_4, & x_5, & x_6 & \geq & 0 \end{array} \tag{A.21}$$

Por ello, el sistema A.20 tiene solución si y sólo si

$$\begin{pmatrix} p \\ q \\ r \end{pmatrix} \in \mathbf{A}_\pi = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 0 \\ 2 & 1 & 0 & 0 & 1 & 0 \\ 0 & -1 & 1 & 0 & 0 & 1 \end{pmatrix}_\pi$$

es decir, si el vector de los términos independientes pertenece al cono generado por las columnas de la matriz de coeficientes. Para saber si ésto se cumple, se obtiene el cono dual del cono $\mathbf{A}_\pi$, usando el algoritmo Γ. La tabla A.6 muestra las tablas que resultan, y que el dual es

$$\mathbf{A}_\pi^* = \mathbf{W}_\pi = \begin{pmatrix} -1 & 0 & 0 \\ 0 & -1 & -1 \\ 0 & 0 & -1 \end{pmatrix}$$

Según (A.19), el sistema es compatible si y sólo si

$$\begin{pmatrix} -1 & 0 & 0 \\ 0 & -1 & -1 \\ 0 & 0 & -1 \end{pmatrix}^T \begin{pmatrix} p \\ q \\ r \end{pmatrix} = \begin{pmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & -1 & -1 \end{pmatrix} \begin{pmatrix} p \\ q \\ r \end{pmatrix} = \begin{pmatrix} -p \\ -q \\ -q-r \end{pmatrix} \leq \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

que conduce a $p \geq 0,\ q \geq 0$ y $r \geq -q$.

■

A.5 Resolución de sistemas lineales

En esta sección se utiliza el Teorema A.2 y el Algoritmo A.1 para resolver sistemas de inecuaciones de la forma

$$\mathbf{Hx} \leq \mathbf{a}$$

Tabla A.6: Ilustración de la aplicación del algoritmo Γ al Ejemplo A.4

Fase 1				Fase 2				Fase 3			
$\mathbf{a}_1$	$\mathbf{v}_1^1$	$\mathbf{v}_2^1$	$\mathbf{v}_3^1$	$\mathbf{a}_2$	$\mathbf{w}_1^2$	$\mathbf{v}_1^2$	$\mathbf{v}_2^2$	$\mathbf{a}_3$	$\mathbf{w}_1^3$	$\mathbf{w}_2^3$	$\mathbf{v}_1^3$
1	**1**	0	0	0	–1	**–2**	0	0	–1	2	**–2**
2	**0**	1	0	1	0	**1**	0	0	0	–1	**1**
0	**0**	0	1	–1	0	**0**	1	1	0	0	**1**
$\mathbf{t}^1$	**1**	2	0	$\mathbf{t}^2$	0	**1**	–1	$\mathbf{t}^3$	0	0	**1**
A_0			1	A_0	2	1	1	A_0	2	1	1
									3	3	2

Fase 4				Fase 5				Fase 6			
$\mathbf{a}_4$	$\mathbf{w}_1^4$	$\mathbf{w}_2^4$	$\mathbf{w}_3^4$	$\mathbf{a}_5$	$\mathbf{w}_1^5$	$\mathbf{w}_2^5$	$\mathbf{w}_3^5$	$\mathbf{a}_6$	$\mathbf{w}_1^6$	$\mathbf{w}_2^6$	$\mathbf{w}_3^6$
1	–1	2	2	0	–1	0	0	0	–1	0	0
0	0	–1	–1	1	0	–1	–1	0	0	–1	–1
0	0	0	–1	0	0	0	–1	1	0	0	–1
$\mathbf{t}^4$	–1	2	2	$\mathbf{t}^5$	0	–1	–1	$\mathbf{t}^6$	0	0	–1
A_0	2	1	1	A_0	2	3	2	A_0	2	3	2
	3	3	2		3	4	4		3	4	4
					5				5	6	
									6		

Más precisamente, dado el poliedro

$$S = \{\mathbf{x}|\mathbf{H}\mathbf{x} \le \mathbf{a}\}$$

se construye, en primer lugar, el cono

$$C_S = \mathbf{C}_\pi^* = \left\{ \begin{pmatrix} \mathbf{x} \\ x_{n+1} \end{pmatrix} |\mathbf{H}\mathbf{x} - x_{n+1}\mathbf{a} \le \mathbf{0};\ -x_{n+1} \le 0 \right\}$$

que es el cono dual del cono generado por las filas de la matriz

$$\mathbf{C} = \begin{bmatrix} \mathbf{H} & | & -\mathbf{a} \\ -- & + & -- \\ 0 & | & -1 \end{bmatrix}$$

En otras palabras, se comienza por escribir el sistema en la forma más conveniente

$$\begin{aligned} \mathbf{c}_i^T\mathbf{x} &\le \mathbf{0}, \forall i \\ x_n &= 1 \end{aligned} \tag{A.22}$$

donde se ha supuesto que $\mathbf{x}$ incluye x_{n+1} y que se han añadido las restricciones redundantes $-x_n \le 0$ a $\mathbf{c}_i^T\mathbf{x} \le \mathbf{0}$. Esto facilita notablemente la obtención de la solución posteriormente.

Es importante hacer notar que siempre es posible escribir un sistema dado en esta forma sin más que transformar:

- Todas las relaciones del tipo $\mathbf{c}_i^T\mathbf{x} \leq \mathbf{a}_i$ por medio de

$$\mathbf{c}_i^T\mathbf{x} - x_n\mathbf{a}_i \leq \mathbf{0} \text{ y } x_n \geq 0 \text{ y } x_n = 1$$

- Las variables $x_i \geq 0$ mediante $\mathbf{c}_i^T\mathbf{x} \leq 0$ con $\mathbf{c}_i = (0, \ldots, -1, \ldots, 0)^T$ con el -1 en la posición i-ésima.

Si $\mathbf{X}$ es el conjunto de las soluciones de (A.22), la definición de cono dual conduce a

$$\mathbf{X} \equiv \{\mathbf{x} \in \mathbf{C}_\pi^* | x_n = 1\} \tag{A.23}$$

Esto implica que se puede encontrar primero el cono dual de $\mathbf{C}_\pi^*$ asociado al primer conjunto de restricciones de (A.22) y luego añadir la última condición.

Utilizando el algoritmo Γ, se obtienen los generadores de C_S y se escribe $C_S = \mathbf{V}_\rho + \mathbf{W}_\pi$. Seguidamente, se identifican los generadores de $\mathbf{W}$ cuya última componente es nula, que se renombran como $\mathbf{W}^*$, y el resto $\mathbf{Q}$. Finalmente, el Teorema A.2 permite escribir

$$S = \mathbf{V}_\rho + \mathbf{W}_\pi^* + \mathbf{Q}_\lambda$$

es decir, la solución de $\mathbf{Hx} \leq \mathbf{a}$.

En resumen, el sistema de ecuaciones e inecuaciones lineales (A.22) puede ser resuelto mediante las etapas siguientes:

1. Se obtiene el cono dual $(\mathbf{B}_\rho + \mathbf{C})_\pi^* = \mathbf{V}_\rho + \mathbf{W}_\pi$
2. Se normalizan los vectores de $\mathbf{W}$ cuya última componente w_n es no nula, dividiendo por ella.
3. Se escribe $\mathbf{C}_\pi^*$ como $\mathbf{V}_\rho + \mathbf{W}_\pi^* + \mathbf{Q}_\lambda$, donde $\mathbf{W}^*$ y $\mathbf{Q}$ son los vectores de $\mathbf{W}$ cuya última componente es nula o unidad, respectivamente

Ejemplo A.5 (sistema de inecuaciones). Considérese el sistema (A.20) con $p = q = r = 1$:

$$\begin{array}{rrrcl} x_1 & & & \leq & 1 \\ 2x_1 & +x_2 & & \leq & 1 \\ & -x_2 & +x_3 & \leq & 1 \\ x_1, & x_2, & x_3 & \geq & 0 \end{array} \tag{A.24}$$

Para resolver este sistema, se escribe en la forma

$$\begin{array}{rrrrcl} x_1 & & & -x_4 & \leq & 0 \\ 2x_1 & +x_2 & & -x_4 & \leq & 0 \\ & -x_2 & +x_3 & -x_4 & \leq & 0 \\ -x_1 & & & & \leq & 0 \\ & -x_2 & & & \leq & 0 \\ & & -x_3 & & \leq & 0 \\ & & & -x_4 & \leq & 0 \\ & & & x_4 & = & 1 \end{array} \tag{A.25}$$

Seguidamente, se resuelve el sistema (A.25) usando las tres etapas indicadas, como sigue.

1. Se obtiene el dual del cono asociado al sistema homogéneo en (A.25). Utilizando el algoritmo Γ, se obtiene la tabla A.7, que en su última tabla da el dual deseado:

$$\pi_1\begin{pmatrix}0\\1\\2\\1\end{pmatrix}+\pi_2\begin{pmatrix}0\\0\\1\\1\end{pmatrix}+\pi_3\begin{pmatrix}1\\0\\2\\2\end{pmatrix}+\pi_4\begin{pmatrix}0\\1\\0\\1\end{pmatrix}+\pi_5\begin{pmatrix}0\\0\\0\\1\end{pmatrix}+\pi_6\begin{pmatrix}1\\0\\0\\2\end{pmatrix}$$

2. Se normalizan estos vectores (dividiendo por su última componente) y se obtiene

$$\pi_1\begin{pmatrix}0\\1\\2\\1\end{pmatrix}+\pi_2\begin{pmatrix}0\\0\\1\\1\end{pmatrix}+\pi_3\begin{pmatrix}\frac{1}{2}\\0\\1\\1\end{pmatrix}+\pi_4\begin{pmatrix}0\\1\\0\\1\end{pmatrix}+\pi_5\begin{pmatrix}0\\0\\0\\1\end{pmatrix}+\pi_6\begin{pmatrix}\frac{1}{2}\\0\\0\\1\end{pmatrix}$$

3. Se escribe el poliedro, que en este caso es el politopo

$$\begin{pmatrix}x_1\\x_2\\x_3\end{pmatrix}=\lambda_1\begin{pmatrix}0\\1\\2\end{pmatrix}+\lambda_2\begin{pmatrix}0\\0\\1\end{pmatrix}+\lambda_3\begin{pmatrix}\frac{1}{2}\\0\\1\end{pmatrix}+\lambda_4\begin{pmatrix}0\\1\\0\end{pmatrix}+\lambda_5\begin{pmatrix}0\\0\\0\end{pmatrix}+\lambda_6\begin{pmatrix}\frac{1}{2}\\0\\0\end{pmatrix}$$

con $\sum_{i=0}^{6}\lambda_i = 1$.

■

Ejemplo A.6 (sistema de inecuaciones). Considérese el siguiente sistema de inecuaciones:

$$\begin{array}{rrrcr} x_1 & & & \leq & -1\\ 2x_1 & +x_2 & & \leq & 0\\ & -x_2 & +x_3 & \leq & -2\\ & -x_2 & +x_3 & \leq & -1\\ x_1 & +x_2 & +x_3 & \leq & -1 \end{array} \tag{A.26}$$

que usando una variable auxiliar se puede escribir como

$$\begin{array}{rrrrcr} x_1 & & & +x_4 & \leq & 0\\ 2x_1 & +x_2 & & & \leq & 0\\ & -x_2 & +x_3 & +2x_4 & \leq & 0\\ & -x_2 & +x_3 & +x_4 & \leq & 0\\ x_1 & +x_2 & +x_3 & +x_4 & \leq & 0\\ & & & -x_4 & \leq & 0\\ & & & x_4 & = & 1 \end{array} \tag{A.27}$$

Puesto que los vectores asociados a las filas de este sistema coinciden con las de la matriz **A** del Ejemplo A.3, se pueden utilizar los resultados anteriores. Por ello, de la tabla A.5 se concluye que la solución es:

$$\begin{pmatrix}x_1\\x_2\\x_3\end{pmatrix}=\lambda_1\begin{pmatrix}-1\\1\\-1\end{pmatrix}+\lambda_2\begin{pmatrix}-1\\2\\-2\end{pmatrix}+\pi_1\begin{pmatrix}0\\-1\\-1\end{pmatrix}+\pi_2\begin{pmatrix}-2\\1\\1\end{pmatrix}+\pi_3\begin{pmatrix}0\\0\\-3\end{pmatrix}+\pi_4\begin{pmatrix}-3\\6\\-3\end{pmatrix}$$

■

Tabla A.7: Tablas correspondientes al algoritmo Γ del Ejemplo A.5

Fase 1				
$\mathbf{a}_1$	$\mathbf{v}_1^1$	$\mathbf{v}_2^1$	$\mathbf{v}_3^1$	$\mathbf{v}_4^1$
1	**1**	0	0	0
0	**0**	1	0	0
0	**0**	0	1	0
–1	**0**	0	0	1
$\mathbf{t}^1$	**1**	0	0	–1
A_0		1	1	

Fase 2				
$\mathbf{a}_2$	$\mathbf{w}_1^2$	$\mathbf{v}_1^2$	$\mathbf{v}_2^2$	$\mathbf{v}_3^2$
2	–1	**0**	0	1
1	0	**1**	0	0
0	0	**0**	1	0
–1	0	**0**	0	1
$\mathbf{t}^2$	–2	**1**	0	1
A_0		1	1 2	1

Fase 3				
$\mathbf{a}_3$	$\mathbf{w}_1^3$	$\mathbf{w}_2^3$	$\mathbf{v}_1^3$	$\mathbf{v}_2^3$
0	–1	0	**0**	1
–1	2	–1	**0**	–1
1	0	0	**1**	0
–1	0	0	**0**	1
$\mathbf{t}^3$	–2	1	**1**	0
A_0	2	1	1 2	1 2 3

Fase 4				
$\mathbf{a}_4$	$\mathbf{w}_1^4$	$\mathbf{w}_2^4$	$\mathbf{w}_3^4$	$\mathbf{v}_1^4$
–1	–1	0	0	**1**
0	2	–1	0	**–1**
0	2	–1	–1	**0**
0	0	0	0	**1**
$\mathbf{t}^4$	1	0	0	**–1**
A_0	2 3	1 3 4	1 2 4	1 2 3

Fase 5				
$\mathbf{a}_5$	$\mathbf{w}_1^5$	$\mathbf{w}_2^5$	$\mathbf{w}_3^5$	$\mathbf{w}_4^5$
0	0	0	0	1
–1	1	–1	0	–1
0	2	–1	–1	0
0	1	0	0	1
$\mathbf{t}^5$	–1	1	0	1
A_0	2 3 4	1 3 4	1 2 4 5	1 2 3

Fase 6				
$\mathbf{a}_6$	$\mathbf{w}_1^6$	$\mathbf{w}_2^6$	$\mathbf{w}_3^6$	$\mathbf{w}_4^6$
0	0	0	0	1
0	0	1	0	0
–1	–1	2	1	2
0	0	1	1	2
$\mathbf{t}^6$	1	–2	–1	–2
A_0	1 2 4 5	2 3 4	3 4 5	2 3 5

Fase 7 y Dual						
$\mathbf{a}_7$	$\mathbf{w}_1^7$	$\mathbf{w}_2^7$	$\mathbf{w}_3^7$	$\mathbf{w}_4^7$	$\mathbf{w}_5^7$	$\mathbf{w}_6^7$
0	0	0	1	0	0	1
0	1	0	0	1	0	0
0	2	1	2	0	0	0
–1	1	1	2	1	1	2
$\mathbf{t}^7$	–1	–1	–2	–1	–1	–2
A_0	2 3 4	3 4 5	2 3 5	2 4 6	4 5 6	2 5 6

A.6 Aplicaciones a varios ejemplos

En esta sección se aplican los métodos desarrollados en las secciones anteriores a varios ejemplos discutidos en los capítulos previos. En particular, se discute el problema de la compatibilidad y se obtienen los conjuntos de todas las soluciones factibles.

A.6.1 El problema del transporte

En esta sección se analiza la compatibilidad y el conjunto de todas las soluciones factibles del problema del transporte discutido en la Sección 1.2. Este problema consiste en la minimización de

$$Z = \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij}$$

sujeto a

$$\begin{array}{rcl} \sum\limits_{j=1}^{n} x_{ij} & = & u_i;\ i = 1, \ldots, m \\ \sum\limits_{i=1}^{m} x_{ij} & = & v_j;\ j = 1, \ldots, n \\ x_{ij} & \geq & 0;\ i = 1, \ldots, m;\ j = 1, \ldots, n \end{array} \tag{A.28}$$

Compatibilidad

Puesto que el conjunto de restricciones (A.28) ya está en la forma (A.17):

$$\begin{array}{rcl} \mathbf{Cx} & = & \mathbf{a} \\ \mathbf{x} & \geq & \mathbf{0} \end{array} \tag{A.29}$$

sólo hay que calcular el dual $\mathbf{C}_\pi^{T^*} = \mathbf{V}_\rho + \mathbf{W}_\pi$ del cono $\mathbf{C}_\pi^T$ y seguidamente comprobar las condiciones $\mathbf{V}^T\mathbf{a} = \mathbf{0}$ y $\mathbf{W}^T\mathbf{a} \leq \mathbf{0}$, donde $\mathbf{a} = \begin{pmatrix} \mathbf{u} \\ \mathbf{v} \end{pmatrix}$. Usando este proceso, se puede mostrar que las condiciones de compatibilidad del sistema (1.2) son

$$\sum_{i=1}^{m} u_i = \sum_{j=1}^{n} v_j;\quad u_i, v_j \geq 0,\quad i = 1, 2, \ldots, m;\quad j = 1, 2, \ldots, n \tag{A.30}$$

que establecen que las cantidades enviadas de todos los orígenes deben ser iguales al total de las cantidades recibidas en todos los destinos, y que estas cantidades deben ser no negativas.

El Conjunto de Todas las Soluciones Factibles

Para obtener el conjunto de todas las soluciones posibles, es necesario escribir

el sistema (A.28) en la forma homogénea equivalente siguiente:

$$\begin{array}{rcl} c_{11}x_{11}+c_{12}x_{12}+\ldots+c_{1n}x_{1n}-a_1y & = & 0 \\ c_{21}x_{21}+c_{22}x_{22}+\ldots+c_{2n}x_{2n}-a_2y & = & 0 \\ \vdots \quad \vdots \quad \vdots \quad \vdots \quad \vdots & = & \vdots \\ c_{m1}x_{m1}+c_{m2}x_{m2}+\ldots+c_{mn}x_{mn}-a_ny & = & 0 \\ -x_{11} & \leq & 0 \\ -x_{12} & \leq & 0 \\ \vdots & \vdots & \vdots \\ -x_{mn} & \leq & 0 \\ -y & \leq & 0 \\ y & = & 1 \end{array} \tag{A.31}$$

De (A.28), se ve que la solución de (A.31) se puede obtener seleccionando los vectores que satisfacen $y = 1$ y pertenecen al cono dual del cono $\mathbf{V}_\rho + \mathbf{W}_\pi$, donde las columnas de $\mathbf{V}$ son los coeficientes de las primeras m filas de (A.31), y las columnas de $\mathbf{W}$ son los coeficientes de las filas $m+1$ a $m+mn+1$ de (A.31).

Ejemplo A.7 (problema del Transporte). Seguidamente, se discute el problema (1.4)

$$\mathbf{Cx} = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_{11} \\ x_{12} \\ x_{13} \\ x_{21} \\ x_{22} \\ x_{23} \\ x_{31} \\ x_{32} \\ x_{33} \end{pmatrix} = \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ v_1 \\ v_2 \\ v_3 \end{pmatrix} \tag{A.32}$$

$$x_{ij} \geq 0; \quad i,j = 1,2,3$$

Compatibilidad

Para obtener las condiciones de compatibilidad asociadas a este problema, es necesario obtener el cono dual del cono generado por las columnas de $\mathbf{C}$. Este

cono es

$$\begin{array}{rcl} \mathbf{C}^*_\pi &=& \mathbf{V}_\rho + \mathbf{W}_\pi \\ &=& \rho \begin{pmatrix} -1 \\ -1 \\ -1 \\ 1 \\ 1 \\ 1 \end{pmatrix} + \pi_1 \begin{pmatrix} -1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} + \pi_2 \begin{pmatrix} 0 \\ -1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} + \pi_3 \begin{pmatrix} 0 \\ 0 \\ -1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \\ && +\pi_4 \begin{pmatrix} 0 \\ 0 \\ 0 \\ -1 \\ 0 \\ 0 \end{pmatrix} + \pi_5 \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ -1 \\ 0 \end{pmatrix} + \pi_6 \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ -1 \end{pmatrix}; \quad \rho \in \mathbb{R}; \quad \pi_i \geq 0; \quad \forall i \end{array} \tag{A.33}$$

que, según (A.19) conduce a las condiciones de compatibilidad

$$\begin{array}{rcl} \mathbf{V}^T\mathbf{a} = 0 & \equiv & -u_1 - u_2 - u_3 + v_1 + v_2 + v_3 = 0 \\ \mathbf{W}^T\mathbf{a} \leq 0 & \equiv & u_1, u_2, u_3, v_1, v_2, v_3 \geq 0 \end{array} \tag{A.34}$$

que coinciden con (A.30).

El Conjunto de todas las Soluciones Factibles

Para encontrar el conjunto de todas las soluciones factibles de (A.32) con $u_1 = 2, u_2 = 3, u_3 = 4; v_1 = 5, v_2 = 2, v_3 = 2$, se escribe el conjunto de restricciones (A.32) como sigue:

$$\begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & -2 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & -3 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & -4 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & -5 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & -2 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & -2 \end{pmatrix} \begin{pmatrix} x_{11} \\ x_{12} \\ x_{13} \\ x_{21} \\ x_{22} \\ x_{23} \\ x_{31} \\ x_{32} \\ x_{33} \\ y \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$\begin{array}{rcl} -x_{ij} & \leq & 0; \quad i, j = 1, 2, 3 \\ -y & \leq & 0 \\ y & = & 1 \end{array} \tag{A.35}$$

Por ello, es necesario obtener el dual del cono $\mathbf{V}_\rho + \mathbf{W}_\pi$, donde las filas de $\mathbf{V}$ son los coeficientes las primeras 6 ecuaciones de (A.35), y las columnas de $\mathbf{W}$ son los coeficientes de las ecuaciones 7–16 de (A.35).

Este dual, que se ha obtenido usando el algoritmo Γ, es

$$\begin{pmatrix} x_{11}\\ x_{12}\\ x_{13}\\ x_{21}\\ x_{22}\\ x_{23}\\ x_{31}\\ x_{32}\\ x_{33}\\ y \end{pmatrix} = \pi_1 \begin{pmatrix} 0\\0\\2\\3\\0\\0\\2\\2\\0\\1 \end{pmatrix} + \pi_2 \begin{pmatrix} 0\\0\\2\\1\\2\\0\\4\\0\\0\\1 \end{pmatrix} + \pi_3 \begin{pmatrix} 0\\2\\0\\3\\0\\0\\2\\0\\2\\1 \end{pmatrix} + \pi_4 \begin{pmatrix} 2\\0\\0\\3\\0\\0\\0\\2\\2\\1 \end{pmatrix} + \pi_5 \begin{pmatrix} 2\\0\\0\\0\\2\\1\\3\\0\\1\\1 \end{pmatrix} + \pi_6 \begin{pmatrix} 2\\0\\0\\1\\2\\0\\2\\0\\2\\1 \end{pmatrix}$$

$$+\pi_7 \begin{pmatrix} 0\\2\\0\\1\\0\\2\\4\\0\\0\\1 \end{pmatrix} + \pi_8 \begin{pmatrix} 2\\0\\0\\1\\0\\2\\2\\2\\0\\1 \end{pmatrix} + \pi_9 \begin{pmatrix} 1\\1\\0\\0\\1\\2\\4\\0\\0\\1 \end{pmatrix} + \pi_{10} \begin{pmatrix} 2\\0\\0\\0\\1\\2\\3\\1\\0\\1 \end{pmatrix} + \pi_{11} \begin{pmatrix} 1\\0\\1\\0\\2\\1\\4\\0\\0\\1 \end{pmatrix} \tag{A.36}$$

donde $\pi_i \geq 0; \quad i = 1, 2, \ldots, 11$. Finalmente, forzando $y = 1$ y eliminando la última componente (asociada a la variable artificial y) de todos los vectores, conduce al politopo:

$$\begin{pmatrix} x_{11}\\ x_{12}\\ x_{13}\\ x_{21}\\ x_{22}\\ x_{23}\\ x_{31}\\ x_{32}\\ x_{33} \end{pmatrix} = \lambda_1 \begin{pmatrix} 0\\0\\2\\3\\0\\0\\2\\2\\0 \end{pmatrix} + \lambda_2 \begin{pmatrix} 0\\0\\2\\1\\2\\0\\4\\0\\0 \end{pmatrix} + \lambda_3 \begin{pmatrix} 0\\2\\0\\3\\0\\0\\2\\0\\2 \end{pmatrix} + \lambda_4 \begin{pmatrix} 2\\0\\0\\3\\0\\0\\0\\2\\2 \end{pmatrix} + \lambda_5 \begin{pmatrix} 2\\0\\0\\0\\2\\1\\3\\0\\1 \end{pmatrix} + \lambda_6 \begin{pmatrix} 2\\0\\0\\1\\2\\0\\2\\0\\2 \end{pmatrix}$$

$$+\lambda_7 \begin{pmatrix} 0\\2\\0\\1\\0\\2\\4\\0\\0 \end{pmatrix} + \lambda_8 \begin{pmatrix} 2\\0\\0\\1\\0\\2\\2\\2\\0 \end{pmatrix} + \lambda_9 \begin{pmatrix} 1\\1\\0\\0\\1\\2\\4\\0\\0 \end{pmatrix} + \lambda_{10} \begin{pmatrix} 2\\0\\0\\0\\1\\2\\3\\1\\0 \end{pmatrix} + \lambda_{11} \begin{pmatrix} 1\\0\\1\\0\\2\\1\\4\\0\\0 \end{pmatrix} \tag{A.37}$$

$$\begin{array}{rcl} \lambda_i & \geq & 0 \\ \sum_{i=1}^{11} \lambda_i & = & 1 \end{array} \tag{A.38}$$

que es el conjunto de todas las soluciones posibles. Nótese que cada columna es una solución correspondiente a un punto extremo del conjunto factible.

De hecho, la solución óptima obtenida en (1.6) puede obtenerse de (A.37). Para ello, se calcula el valor de la función objetivo (1.5) para la solución general de (A.37) y se obtiene

$$\begin{pmatrix} 1 & 2 & 3 & 2 & 1 & 2 & 3 & 2 & 1 \end{pmatrix} \begin{pmatrix} x_{11} \\ x_{12} \\ x_{13} \\ x_{21} \\ x_{22} \\ x_{23} \\ x_{31} \\ x_{32} \\ x_{33} \end{pmatrix}$$

$$= 22\lambda_1 + 22\lambda_2 + 18\lambda_3 + 14\lambda_4 + 16\lambda_5 + 14\lambda_6 + 22\lambda_7 + 18\lambda_8 + 20\lambda_9 + 18\lambda_{10} + 20\lambda_{11}. \tag{A.39}$$

La expresión (A.39) contiene todos los valores posibles de la función objetivo. Para obtener un valor posible basta distribuir una unidad entre los 11 valores λ. Puesto que hay muchas opciones posibles para hacer esta distribución, y se busca minimizar este valor, se buscan los λ que tengan el mínimo coeficiente. Éstas son λ_4 y λ_6, que comparten el coeficiente 14. Entonces, se distribuye la unidad entre ellos dos y se hacen nulos los restantes λ. En consecuencia, de (A.39) se concluye que el mínimo valor de la función objetivo es 14, que corresponde a cualquier combinación lineal convexa de los puntos asociados a λ_4 y λ_6, puesto que ambos tienen un coeficiente de valor 14. Por ello, el conjunto de todas las soluciones, es decir, las que conducen a un valor 14 de la función objetivo, es

$$\begin{pmatrix} x_{11} \\ x_{12} \\ x_{13} \\ x_{21} \\ x_{22} \\ x_{23} \\ x_{31} \\ x_{32} \\ x_{33} \end{pmatrix} = \lambda \begin{pmatrix} 2 \\ 0 \\ 0 \\ 3 \\ 0 \\ 0 \\ 0 \\ 2 \\ 2 \end{pmatrix} + (1-\lambda) \begin{pmatrix} 2 \\ 0 \\ 0 \\ 1 \\ 2 \\ 0 \\ 2 \\ 0 \\ 2 \end{pmatrix} \; ; \quad 0 \leq \lambda \leq 1 \tag{A.40}$$

que, por supuesto, incluye la solución $(2, 0, 0, 1, 2, 0, 2, 0, 2)^T$, que se dió en (1.6), de la Sección 1.6. ■

A.6.2 El problema de la planificación de la producción

En esta sección se analiza la compatibilidad y el conjunto de todas las soluciones del problema de la planificación de la producción que se discutió en la Sección 1.3. Este problema consiste en minimizar

$$\sum_{t=1}^{n}(a_t y_t - b_t x_t - c_t s_t)$$

sujeto a

$$\begin{array}{lcl} s_{t-1} + x_t - s_t & = & y_t, \;\; t = 1, 2, \ldots, n \\ s_t, x_t, y_t & \geq & 0 \end{array} \tag{A.41}$$

Compatibilidad

Puesto que (A.41) es un sistema de ecuaciones en variables no negativas, se pueden utilizar los resultados de la Sección A.4 directamente para discutir el problema de la compatibilidad. Nuestro sistema se convierte en

$$\begin{array}{rcl} x_1 - s_1 & = & y_1 - s_0 \\ s_1 + x_2 - s_2 & = & y_2 \\ \vdots & \vdots & \vdots \\ s_{n-1} + x_n - s_n & = & y_n \\ s_i, x_i & \geq & 0; \;\; i = 1, 2, \ldots, n \end{array} \tag{A.42}$$

Para analizar su compatibilidad, es necesario obtener el dual $\mathbf{C}_\pi^* = \mathbf{V}_\rho + \mathbf{W}_\pi$ del cono correspondiente $\mathbf{C}_\pi$ y seguidamente comprobar las condiciones $\mathbf{V}^T\mathbf{a} = \mathbf{0}$ and $\mathbf{W}^T\mathbf{a} \leq \mathbf{0}$, donde $\mathbf{a} = (y_1 - s_0, y_2, \ldots, y_n)^T$.

Se puede demostrar que el cono dual es vacío, y por ello, el problema es siempre compatible.

El Conjunto de Todas las Soluciones Factibles

Para obtener el conjunto de todas las soluciones posibles es necesario escribir el sistema (A.42) en la forma homogénea equivalente:

$$\begin{array}{rcl} x_1 - s_1 - (y_1 - s_0)x_{n+1} & = & 0 \\ s_1 + x_2 - s_2 - y_2 x_{n+1} & = & 0 \\ \vdots & \vdots & \vdots \\ s_{n-1} + x_n - s_n - y_n x_{n+1} & = & 0 \\ -s_i & \leq & 0; \;\; i = 1, 2, \ldots, n \\ -x_i & \leq & 0; \;\; i = 1, 2, \ldots, n \\ -x_{n+1} & \leq & 0 \\ x_{n+1} & = & 1 \end{array} \tag{A.43}$$

Por ello, la solución de (A.42) puede obtenerse seleccionando los vectores que satisfacen $x_{n+1} = 1$ y pertenecen al cono dual del cono $\mathbf{V}_\rho + \mathbf{W}_\pi$, donde

las columnas de $\mathbf{V}$ son los coeficientes de las primeras n ecuaciones de (A.43), y las columnas de $\mathbf{W}$ son los coeficientes de las ecuaciones $n+1$ a $3n+1$ de (A.43).

Ejemplo A.8 (problema de la planificación de la producción). Para obtener las condiciones de compatibilidad del sistema (A.41), es necesario obtener el cono dual del cono generado por sus columnas. Este cono es vacío. Por ello, el problema es siempre factible para cualquier función de demanda y cualquier valor almacenado inicial s_0.

Supóngase ahora que se tiene un límite de almacenamiento s_t^{max} en el instante t. Entonces, introduciendo cuatro variables auxiliares de holgura no negativas z_1, z_2, z_3, z_4, el sistema (1.10) se transforma en

$$\mathbf{Cx} = \begin{pmatrix} -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} s_1 \\ s_2 \\ s_3 \\ s_4 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ z_1 \\ z_2 \\ z_3 \\ z_4 \end{pmatrix} = \begin{pmatrix} 0 \\ 3 \\ 6 \\ 1 \\ s_1^{max} \\ s_2^{max} \\ s_3^{max} \\ s_4^{max} \end{pmatrix}$$

$$s_t, x_t, z_t \geq 0 \tag{A.44}$$

En este caso, el dual del cono generado por las columnas de $\mathbf{C}$ es el cono generado por las columnas de la matriz $\mathbf{W}$:

$$\begin{pmatrix} 0 & 0 & -1 & 0 & 0 & -1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & -1 & -1 & 0 & 0 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & -1 & 0 & -1 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & -1 & -1 \\ 0 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & -1 & -1 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & -1 & -1 \end{pmatrix} \tag{A.45}$$

que conduce a las condiciones siguientes:

$$\begin{array}{rcl}
s_4^{max} & \geq & 0 \\
s_1^{max} & \geq & 0 \\
-s_0 + s_1^{max} + y_1 & \geq & 0 \\
s_2^{max} & \geq & 0 \\
s_2^{max} + y_2 & \geq & 0 \\
-s_0 + s_2^{max} + y_1 + y_2 & \geq & 0 \\
s_3^{max} & \geq & 0 \\
s_3^{max} + y_3 & \geq & 0 \\
s_3^{max} + y_2 + y_3 & \geq & 0 \\
-s_0 + s_3^{max} + y_1 + y_2 + y_3 & \geq & 0 \\
s_4^{max} + y_4 & \geq & 0 \\
s_4^{max} + y_3 + y_4 & \geq & 0 \\
s_4^{max} + y_2 + y_3 + y_4 & \geq & 0 \\
-s_0 + s_4^{max} + y_1 + y_2 + y_3 + y_4 & \geq & 0
\end{array}$$

donde s_t^{max} es la capacidad de almacenamiento en el instante t.

El Conjunto de todas las Soluciones Factibles

Para encontrar el conjunto de todas las soluciones posible de (1.10), se escribe el conjunto de restricciones en la forma

$$\begin{pmatrix} -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 & 1 & 0 & 0 & -3 \\ 0 & 1 & -1 & 0 & 0 & 0 & 1 & 0 & -6 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 1 & -1 \end{pmatrix} \begin{pmatrix} s_1 \\ s_2 \\ s_3 \\ s_4 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ y \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \qquad \text{(A.46)}$$

y

$$\begin{array}{rcl}
-s_i & \leq & 0; \;\; i = 1,2,3,4 \\
-x_i & \leq & 0; \;\; i = 1,2,3,4 \\
-y & \leq & 0 \\
y & = & 1
\end{array} \qquad \text{(A.47)}$$

Por ello, es necesario obtener el dual del cono $\mathbf{V}_\rho + \mathbf{W}_\pi$, donde las columnas de $\mathbf{V}$ son los coeficientes de las cuatro primeras filas de (A.46)–(A.47), y las columnas de $\mathbf{W}$ son los coeficientes de las filas 1 a 9 de (A.47).

Este dual es $\mathbf{C}_\pi^* = \mathbf{W}_\pi$, donde la matriz $\mathbf{W}$ es la dada en la tabla A.8. Ahora, forzando a las últimas componentes de los vectores de $\mathbf{C}_\pi^*$ a ser unidad ($y = 1$), se obtiene que la solución general $\mathbf{W}_\pi^1 + \mathbf{W}_\lambda^2$ de (1.10) es una combinación lineal no negativa de los vectores de $\mathbf{W}^1$ más una combinación lineal convexa de los vectores de $\mathbf{W}^2$, donde $\mathbf{W}^1$ es la matriz que contiene las cuatro primeras

Tabla A.8: Conjunto de puntos extremos asociados al sistema (1.10).

$\mathbf{w}_1$	$\mathbf{w}_2$	$\mathbf{w}_3$	$\mathbf{w}_4$	$\mathbf{w}_5$	$\mathbf{w}_6$	$\mathbf{w}_7$	$\mathbf{w}_8$	$\mathbf{w}_9$	$\mathbf{w}_{10}$	$\mathbf{w}_{11}$	$\mathbf{w}_{12}$
0	0	$\frac{10}{3}$	0	3	0	9	0	3	0	10	0
0	0	$\frac{10}{3}$	$\frac{10}{3}$	0	0	6	6	0	0	7	7
0	1	$\frac{10}{3}$	$\frac{10}{3}$	0	0	0	0	1	1	1	1
1	1	$\frac{10}{3}$	$\frac{10}{3}$	0	0	0	0	0	0	0	0
0	0	$\frac{10}{3}$	0	3	0	9	0	3	0	10	0
0	0	0	$\frac{10}{3}$	0	3	0	9	0	3	0	10
0	1	0	0	6	6	0	0	7	7	0	0
1	0	0	0	1	1	1	1	0	0	0	0
0	0	0	0	1	1	1	1	1	1	1	1

columnas, y $\mathbf{W}^2$, las columnas restantes de la tabla A.8, es decir, un cono más un politopo.

Nótese que

1. Cada vector de $\mathbf{W}^2$ es una solución de (A.44). Estas soluciones representan la producción mínima requerida para satisfacer a la demanda. Ellas conducen a un almacenamiento nulo en al menos un instante.

2. Cada vector de $\mathbf{W}^1$ es una solución del sistema homogéneo asociado. Estas soluciones representan los excesos de la producción sobre la demanda y por tanto sólo representan incrementos de almacenamiento, haciendo la política de producción más robusta frente a posibles incrementos de demanda.

Si, alternativamente, se busca el conjunto de todas las soluciones de (A.44),

con $s_t^{max} = 4; \forall t$, se escribe el conjunto de restricciones en la forma

$$\begin{pmatrix} -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & -3 \\ 0 & 1 & -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -6 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & -1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & -4 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & -4 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -4 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -4 \end{pmatrix} \begin{pmatrix} s_1 \\ s_2 \\ s_3 \\ s_4 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ z_1 \\ z_2 \\ z_3 \\ z_4 \\ y \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$\begin{array}{rcl} -s_t, -x_t, -z_t & \leq & 0; \ \forall t \\ -y & \leq & 0 \\ y & = & 1 \end{array} \tag{A.48}$$

y se obtiene el dual en la tabla A.9. Por ello, la solución general de (A.44) es la combinación lineal convexa de los vectores de la tabla A.9, con sus últimas componentes eliminadas, es decir, un politopo.

Reemplazando la solución general de la tabla A.8 en la función a maximizar

$$Z = \sum_{i=1}^{n} (a_t d_t - b_t x_t - c_t s_t)$$

con $a_t = 3, b_t = 1, c_t = 1$, y la función de demanda de la tabla 1.1, se obtiene

$$\begin{array}{rcl} Z & = & 36 - 2\pi_1 - 3\pi_2 - \frac{50}{3}\pi_3 - \frac{54}{3}\pi_4 - 13\lambda_1 - 10\lambda_2 - 25\lambda_3 - 16\lambda_4 \\ & & -14\lambda_5 - 11\lambda_6 - 28\lambda_7 - 18\lambda_8. \end{array} \tag{A.49}$$

Puesto que todos los coeficientes π son negativos, y el mínimo coeficiente de λ es -10, y corresponde a λ_2, la solución óptima es única y se obtiene para $\pi_1 = \pi_2 = \pi_3 = \pi_4 = 0$ y $\lambda_i = 0; i \neq 2; \lambda_2 = 1$, que conduce a $Z = 36 - 10 = 26$.

Una opción alternativa consiste en minimizar el almacenamiento, como en (1.9):

$$Z = s_1 + s_2 + s_3 + s_4. \tag{A.50}$$

Calculando el valor de la función a minimizar usando la solución general de la tabla A.8, se obtiene

$$Z = \pi_1 + 2\pi_2 + \frac{40}{3}\pi_3 + 10\pi_4 + 3\lambda_1 + 15\lambda_3 + 6\lambda_4 + 4\lambda_5 + 1\lambda_6 + 18\lambda_7 + 8\lambda_8 \tag{A.51}$$

Puesto que todos los coeficientes π son positivos, y el mínimo coeficiente de los λ es 0, para λ_2, se obtiene un mínimo de valor $Z = 0$, que corresponde a $\pi_1 = \pi_2 = \pi_3 = \pi_4 = 0$ y $\lambda_i = 0; i \neq 2; \lambda_2 = 1$.

Tabla A.9: Conjunto de los puntos extremos asociados al sistema (A.44)

$\mathbf{w}_1$	$\mathbf{w}_2$	$\mathbf{w}_3$	$\mathbf{w}_4$	$\mathbf{w}_5$	$\mathbf{w}_6$	$\mathbf{w}_7$	$\mathbf{w}_8$	$\mathbf{w}_9$	$\mathbf{w}_{10}$	$\mathbf{w}_{11}$	$\mathbf{w}_{12}$	$\mathbf{w}_{13}$
3	0	3	0	4	4	0	0	4	4	3	0	4
0	0	0	0	1	1	4	4	4	4	0	0	1
0	0	1	1	0	1	0	1	0	1	4	4	4
0	0	0	0	0	0	0	0	0	0	3	3	3
3	0	3	0	4	4	0	0	4	4	3	0	4
0	3	0	3	0	0	7	7	3	3	0	3	0
6	6	7	7	5	6	2	3	2	3	10	10	9
1	1	0	0	1	0	1	0	1	0	0	0	0
1	4	1	4	0	0	4	4	0	0	1	4	0
4	4	4	4	3	3	0	0	0	0	4	4	3
4	4	3	3	4	3	4	3	4	3	0	0	0
4	4	4	4	4	4	4	4	4	4	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1

$\mathbf{w}_{14}$	$\mathbf{w}_{15}$	$\mathbf{w}_{16}$	$\mathbf{w}_{17}$	$\mathbf{w}_{18}$	$\mathbf{w}_{19}$	$\mathbf{w}_{20}$	$\mathbf{w}_{21}$	$\mathbf{w}_{22}$	$\mathbf{w}_{23}$	$\mathbf{w}_{24}$	$\mathbf{w}_{25}$
0	4	3	0	4	0	4	3	0	4	0	4
4	4	0	0	1	4	4	0	0	1	4	4
4	4	0	0	0	0	0	4	4	4	4	4
3	3	4	4	4	4	4	4	4	4	4	4
0	4	3	0	4	0	4	3	0	4	0	4
7	3	0	3	0	7	3	0	3	0	7	3
6	6	6	6	5	2	2	10	10	9	6	6
0	0	5	5	5	5	5	1	1	1	1	1
4	0	1	4	0	4	0	1	4	0	4	0
0	0	4	4	3	0	0	4	4	3	0	0
0	0	4	4	4	4	4	0	0	0	0	0
1	1	0	0	0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1	1	1	1

Si en vez de las restricciones (1.10), se usan las (A.44), se obtiene el dual de la tabla A.9 y (A.49) se transforma en

$$
\begin{array}{rcl}
Z & = & 36 - 13\lambda_1 - 10\lambda_2 - 14\lambda_3 - 11\lambda_4 - 15\lambda_5 - 16\lambda_6 - 14\lambda_7 \\
 & = & -15\lambda_8 - 18\lambda_9 - 19\lambda_{10} - 23\lambda_{11} - 20\lambda_{12} - 25\lambda_{13} - 24\lambda_{14} \\
 & = & -28\lambda_{15} - 21\lambda_{16} - 18\lambda_{17} - 23\lambda_{18} - 22\lambda_{19} - 26\lambda_{20} - 25\lambda_{21} - 22\lambda_{22} \\
 & & -27\lambda_{23} - 26\lambda_{24} - 30\lambda_{25}
\end{array} \tag{A.52}
$$

que conduce a un valor máximo de 26 para $\lambda_2 = 1; \lambda_i = 0; \forall i \neq 2$. Este valor, según la tabla A.9, se alcanza en

$$(s_1, s_2, s_3, s_4, x_1, x_2, x_3, x_4) = (0, 0, 0, 0, 0, 3, 6, 1)$$

donde se han usado sólo las ocho primeras componentes de $\mathbf{w}_2$ (se han eliminado

Tabla A.10: Ejemplo de una tabla "input–output".

Ventas	Ferrocarril	Acero	Carbón	Otras	Demanda Final	Total
Ferrocarril	x_{11}	x_{12}	x_{13}	x_{14}	y_1	x_1
Acero	x_{21}	x_{22}	x_{23}	x_{24}	y_2	x_2
Carbón	x_{31}	x_{32}	x_{33}	x_{34}	y_3	x_3
Otras	x_{41}	x_{42}	x_{43}	x_{44}	y_4	x_4

los valores de las variables auxiliares $z_i = 4; i = 1, 2, 3, 4$).

Nótese que ahora la producción está limitada por el límite de almacén. Nótese también que todas las soluciones de la tabla A.8 con sus primeras cuatro componentes inferiores a 4 (el límite de almacén) están incluidas en la tabla A.9. El resto de soluciones de la tabla A.9 tienen al menos una de sus cuatro primeras componentes iguales a 4 (límite de almacén).

Similarmente, con (A.44), (A.51) se transforma en

$$\begin{array}{rcl} M & = & 3\lambda_1 + 4\lambda_3 + 1\lambda_4 + 5\lambda_5 + 6\lambda_6 + 4\lambda_7 + 5\lambda_8 + 8\lambda_9 \\ & & +9\lambda_{10} + 10\lambda_{11} + 7\lambda_{12} + 12\lambda_{13} + 11\lambda_{14} + 15\lambda_{15} + 7\lambda_{16} + 4\lambda_{17} \\ & & +9\lambda_{18} + 8\lambda_{19} + 12\lambda_{20} + 11\lambda_{21} + 8\lambda_{22} + 13\lambda_{23} + 12\lambda_{24} + 16\lambda_{25} \end{array} \quad (A.53)$$

que conduce a un valor mínimo de 0 para $\lambda_2 = 1; \lambda_i = 0; i \neq 2$. Este valor, según la tabla A.9, se alcanza también en

$$(s_1, s_2, s_3, s_4, x_1, x_2, x_3, x_4) = (0, 0, 0, 0, 0, 3, 6, 1)$$

■

A.6.3 Las tablas "input–output"

Como una introducción a las tablas input–output, considérese una economía que consta de tres industrias básicas – ferrocarril, acero y carbón – y una cuarta categoría con el resto de industrias. El objetivo del problema consiste en analizar las interrelaciones de estas industrias en función de las ventas de cada una a las otras y a los demás elementos de la economía, durante un periodo de tiempo dado. Este análisis puede hacerse mediante una tabla input–output. Este problema no cae en la categoría de los problemas de optimización.

Cada elemento de la tabla A.10 representa el total de ventas de cada industria durante el periodo considerado. Por ejemplo, la primera fila de la tabla describe las ventas de la industria del ferrocarril a cada una de las demás industrias.

Por ello, los tres elementos principales del problema de las tablas input–output son:

1. **Datos.**

   x_{ij}: el total de ventas de la industria i a la industria j durante el periodo considerado

Tabla A.11: Ejemplo de una tabla "input–output".

Ventas	Ferrocarril	Acero	Carbón	Otros	Demanda final	Total
Ferrocarril	2	2	6	4	6	20
Acero	5	1	2	4	8	20
Acero	3	4	2	5	6	20
Otros	6	6	3	6	9	30

Tabla A.12: Ejemplo de coeficientes en una tabla "input–output".

Ventas	Ferrocarril	Acero	Carbón	Otros	Demanda final
Ferrocarril	0.10	0.10	0.30	0.20	0.30
Acero	0.25	0.05	0.10	0.20	0.40
Carbón	0.15	0.20	0.10	0.25	0.30
Otros	0.20	0.20	0.10	0.20	0.30

y_i: La demanda final de la industria i

2. **Variables.**

   x_i: la producción total de la industria i durante el periodo considerado

3. **Restricciones.** Las interrelaciones de las industrias en el periodo considerado pueden escribirse como

$$x_i - \sum_{j=1,\ldots,n} x_{ij} = y_i; \quad i = 1, \ldots, n \tag{A.54}$$

   En los años 30 Leontieff supuso que el modelo lineal de una economía para un periodo dado podía utilizarse para predecir la estructura de dicha economía en un periodo futuro, y llamó $a_{ij} = x_{ij}/x_j$, al *coeficiente input–output*, que es la cantidad de industria i necesaria para producir una unidad de producto j. Por ello, el sistema (A.54) resulta

$$\begin{array}{rcl} (1-a_{ii})x_i - \sum\limits_{j \neq i} a_{ij}x_j & = & y_i; \quad i = 1, \ldots, n \\ y_i & \geq & 0; \quad i = 1, \ldots, n \\ x_{ij} & \geq & 0 \end{array} \tag{A.55}$$

Ejemplo A.9 (tablas Input–Output). Considérese la tabla input–output A.11, y sus correspondientes coeficientes en la tabla A.12. En este caso, el sistema (A.55) se transforma en

$$\begin{array}{c} \begin{pmatrix} 0.90 & -0.10 & -0.30 & -0.20 \\ -0.25 & 0.95 & -0.10 & -0.20 \\ -0.15 & -0.20 & 0.90 & -0.25 \\ -0.20 & -0.20 & -0.10 & 0.80 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 6 \\ 8 \\ 6 \\ 9 \end{pmatrix} \\ x_i \geq 0; \quad i = 1, 2, 3, 4. \end{array} \tag{A.56}$$

Tabla A.13: Dual del cono generado por las columnas de la matriz en (A.56)

$\mathbf{w}_1$	$\mathbf{w}_2$	$\mathbf{w}_3$	$\mathbf{w}_4$
–1.481	–0.568	–0.534	–0.579
–0.426	–1.323	–0.504	–0.500
–0.615	–0.403	–1.423	–0.432
–0.669	–0.599	–0.704	–1.655

Compatibilidad

Las condiciones de compatibilidad del sistema (A.56) se obtienen obteniendo el dual del cono generado pos los correspondientes vectores columna, que se dan en la tabla A.13. Por ello, es compatible si y sólo si

$$\begin{pmatrix} -1.481 & -0.426 & -0.615 & -0.669 \\ -0.568 & -1.323 & -0.403 & -0.599 \\ -0.534 & -0.504 & -1.423 & -0.704 \\ -0.579 & -0.500 & -0.432 & -1.655 \end{pmatrix} \begin{pmatrix} d_1 \\ d_2 \\ d_3 \\ d_4 \end{pmatrix} \leq \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \tag{A.57}$$

donde d_1, d_2, d_3, d_4 son las demandas finales de las cuatro industrias.

Nótese que para demandas finales no negativas, el sistema siempre tiene solución.

Si las tablas input–output están correctamente definidas, este problema siempre tiene solución para demandas finales positivas.

El Conjunto de todas las Soluciones Factibles

La solución general del sistema (A.56) es

$$\begin{pmatrix} x_1 & x_2 & x_3 & x_4 \end{pmatrix}^T = \begin{pmatrix} \frac{3125}{142} & \frac{3095}{142} & \frac{1570}{71} & \frac{3545}{142} \end{pmatrix}^T, \tag{A.58}$$

que da la solución única del mismo.

■

A.6.4 El problema de la dieta

Se analiza en esta sección el problema de la dieta.

Ejemplo A.10 (el problema de la dieta). En este ejemplo se estudian las condiciones de compatibilidad y el conjunto de posibles soluciones del problema de la dieta. Se interpretarán físicamente cada una de las componentes que aparecen en la solución general, es decir, las combinaciones lineales no negativas y

convexas; interesantes resultados que se obtienen del análisis de los resultados matemáticos.

Compatibilidad

Utilizando las variables de holgura $z_1, \ldots, z_4$, el sistema (1.14) se transforma en

$$\begin{pmatrix} 78.6 & 70.1 & 80.1 & 67.2 & 77.0 & -1 & 0 & 0 & 0 \\ 6.50 & 9.40 & 8.80 & 13.7 & 30.4 & 0 & -1 & 0 & 0 \\ 0.02 & 0.09 & 0.03 & 0.14 & 0.41 & 0 & 0 & -1 & 0 \\ 0.27 & 0.34 & 0.30 & 1.29 & 0.86 & 0 & 0 & 0 & -1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ z_1 \\ z_2 \\ z_3 \\ z_4 \end{pmatrix} = \begin{pmatrix} 74.2 \\ 14.7 \\ 0.14 \\ 0.55 \end{pmatrix}$$

$$x_1, x_2, x_3, x_4, x_5, z_1, z_2, z_3, z_4 \geq 0. \quad \text{(A.59)}$$

que es compatible puesto que el cono dual asociado a las columnas de la matriz en (A.59) es vacío.

El Conjunto de todas las Soluciones Factibles

El conjunto de soluciones factibles se obtiene a partir del sistema homogéneo asociado a (1.14). Esta solución es $\mathbf{W}_\pi + \mathbf{Q}_\lambda$, donde $\mathbf{W}$ y $\mathbf{Q}$ son las que se muestran en la tabla A.14. Nótese que esta solución general consiste en una combinación lineal no negativa de los vectores de $\mathbf{W}$ más una combinación lineal convexa de los vectores de $\mathbf{Q}$, es decir, un cono más un politopo. El significado de ambas partes es claro:

- $\mathbf{Q}$ contiene soluciones con las cantidades mínimas de nutrientes requeridas para satisfacer los requerimientos mínimos.

- $\mathbf{W}$ contiene cualquier cantidad de nutriente que conduce a un exceso de los requerimientos.

Como en los ejemplos anteriores, el mínimo puede obtenerse de la tabla A.14. ■

A.6.5 El problema de las redes

Puesto que el sistema (1.19) no está en la forma (A.17), es decir, en función de variables no negativas, se definen las nuevas variables

$$y_{ij} = x_{ij} + m_{ij}; \forall i, j$$

Tabla A.14: Vector que genera la solución del ejemplo del problema de la dieta

	Maíz a	Avena	Maíz b	Salvado	Linaza
			W		
$\mathbf{w}_1$	1	0	0	0	0
$\mathbf{w}_2$	0	1	0	0	0
$\mathbf{w}_3$	0	0	1	0	0
$\mathbf{w}_4$	0	0	0	1	0
$\mathbf{w}_5$	0	0	0	0	1
			Q		
$\mathbf{q}_1$	0	0	0	1.10417	0
$\mathbf{q}_2$	0	0	0	0	0.963636
$\mathbf{q}_3$	0.0448423	0	0	1.05172	0
$\mathbf{q}_4$	0	0.0873163	0	1.01308	0
$\mathbf{q}_5$	0	0	0.0567182	1.03656	0
$\mathbf{q}_6$	7.	0	0	0	0
$\mathbf{q}_7$	0	0	4.66667	0	0
$\mathbf{q}_8$	0.220126	0	0	0.968553	0
$\mathbf{q}_9$	0	0	0.170524	0.963459	0
$\mathbf{q}_{10}$	0	1.61765	0	0	0
$\mathbf{q}_{11}$	0.458531	0	0	0	0.495577
$\mathbf{q}_{12}$	0	0.629273	0	0	0.390752
$\mathbf{q}_{13}$	0	0	0.468746	0	0.476019
$\mathbf{q}_{14}$	0.544348	0	0	0.096701	0.323583
$\mathbf{q}_{15}$	0	1.53026	0	0.023032	0
$\mathbf{q}_{16}$	0	0.736664	0	0.088176	0.216031
$\mathbf{q}_{17}$	0	0	0.574136	0.116168	0.265004
$\mathbf{q}_{18}$	0.10857	1.53143	0	0	0
$\mathbf{q}_{19}$	1.12406	0	0	0	0.286631
$\mathbf{q}_{20}$	0	1.51786	0.11309	0	0
$\mathbf{q}_{21}$	0	0	1.08128	0	0.262346
$\mathbf{q}_{22}$	0.022243	1.51631	0	0.022055	0
$\mathbf{q}_{23}$	0.814007	0	0	0.070968	0.277523
$\mathbf{q}_{24}$	0	1.51325	0.017539	0.023436	0
$\mathbf{q}_{25}$	0	0	0.598964	0.114763	0.258449

y el sistema (1.19) se transforma en

$$\begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} y_{12} \\ y_{13} \\ y_{14} \\ y_{24} \\ y_{34} \\ z_{12} \\ z_{13} \\ z_{14} \\ z_{24} \\ z_{34} \end{pmatrix} = \begin{pmatrix} f_1^* \\ f_2^* \\ f_3^* \\ f_4^* \\ 2m_{12} \\ 2m_{13} \\ 2m_{14} \\ 2m_{24} \\ 2m_{34} \end{pmatrix},$$
$$\begin{array}{rcl} y_{ij} & \geq & 0; \forall i,j \\ z_{ij} & \geq & 0; \forall i,j \end{array} \tag{A.60}$$

donde $z_{ij} \geq 0; \quad \forall i, j$ son las variables de holgura auxiliares utilizadas para convertir el sistema de inecuaciones (1.19) en el sistema de ecuaciones (A.60), y

$$\begin{pmatrix} f_1^* \\ f_2^* \\ f_3^* \\ f_4^* \end{pmatrix} = \begin{pmatrix} f_1 + m_{12} + m_{13} + m_{14} \\ f_2 - m_{12} + m_{24} \\ f_3 - m_{13} + m_{34} \\ f_4 - m_{14} - m_{24} - m_{34} \end{pmatrix} \tag{A.61}$$

Compatibilidad

Un análisis de la compatibilidad del sistema requiere que se obtenga el dual $\mathbf{C}_\pi^*$ del cono generado por las columnas de la matriz $\mathbf{C}$ en (A.60), cuyos generadores se dan en la primera columna de la tabla A.15. Puede verse que hay sólo un vector $\mathbf{V}$ y diecisiete $\mathbf{W}$, que conducen a una ecuación de compatibilidad (la primera en la tabla A.15) y 17 inecuaciones (las restantes filas de la tabla A.15).

La ecuación de compatibilidad

$$f_1 + f_2 + f_3 + f_4 = 0 \tag{A.62}$$

dice sólo que la ley de conservación de flujo debe satisfacerse globalmente, es decir, el flujo que entra al sistema y el que sale del mismo deben coincidir (se supone ausencia de pérdidas).

Las condiciones de desigualdad deben dividirse en dos grupos:

Restricciones de desigualdad negativas, que sólo indican que las capacidades deben ser no negativas.

Restricciones de desigualdad de máximo flujo, que requieren que el flujo que entra o sale no supere las capacidades máximas asociadas a las diferentes secciones (indicadas en la Figura A.3). Por ejemplo, en la sección S_1, el flujo máximo está dado por $m_{12} + m_{13} + m_{14}$ (el máximo flujo de las conducciones que concurren

Tabla A.15: Condiciones de compatibilidad.

Vector	Restricción	Sección
Restricciones de igualdad que expresan el balance de flujo		
$(1,1,1,1,0,0,0,0,0)^T$	$f_1+f_2+f_3+f_4=0$	
Restricciones de no negatividad		
$(0,0,0,0,-1,0,0,0,0)^T$	$m_{12}\geq 0$	
$(0,0,0,0,0,-1,0,0,0)^T$	$m_{13}\geq 0$	
$(0,0,0,0,0,0,-1,0,0)^T$	$m_{14}\geq 0$	
$(0,0,0,0,0,0,0,-1,0)^T$	$m_{24}\geq 0$	
$(0,0,0,0,0,0,0,0,-1)^T$	$m_{34}\geq 0$	
Restricciones de máximo flujo posible		
$(-1,0,0,0,0,0,0,0,0)^T$	$-f_1\leq m_{12}+m_{13}+m_{14}$	S_1
$(1,0,0,0,-1,-1,-1,0,0)^T$	$f_1\leq m_{12}+m_{13}+m_{14}$	S_1
$(0,1,0,0,0,0,0,-1,0)^T$	$f_2\leq m_{12}+m_{24}$	S_2
$(0,-1,0,0,-1,0,0,0,0)^T$	$-f_2\leq m_{12}+m_{24}$	S_2
$(0,0,1,0,0,0,0,0,-1)^T$	$f_3\leq m_{13}+m_{34}$	S_3
$(0,0,-1,0,0,-1,0,0,0)^T$	$-f_3\leq m_{13}+m_{34}$	S_3
$(1,1,1,0,0,0,-1,-1,-1)^T$	$f_1+f_2+f_3\leq m_{14}+m_{24}+m_{34}$	S_4
$(-1,-1,-1,0,0,0,0,0,0)^T$	$-f_1-f_2-f_3\leq m_{14}+m_{24}+m_{34}$	S_4
$(1,0,1,0,-1,0,-1,0,-1)^T$	$f_1+f_3\leq m_{12}+m_{14}+m_{34}$	S_5
$(-1,0,-1,0,0,0,0,0,0)^T$	$-f_1-f_3\leq m_{12}+m_{14}+m_{34}$	S_5
$(-1,-1,0,0,0,0,0,0,0)^T$	$-f_1-f_2\leq m_{13}+m_{14}+m_{24}$	S_6
$(1,1,0,0,0,-1,-1,-1,0)^T$	$f_1+f_2\leq m_{13}+m_{14}+m_{24}$	S_6

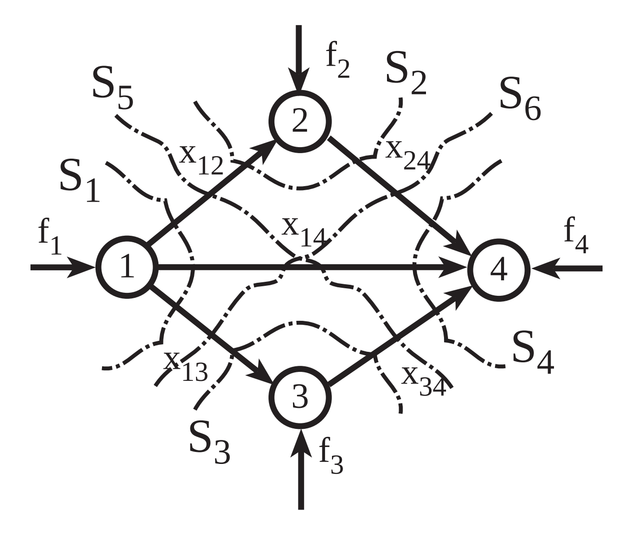


Figura A.3: Secciones asociadas a las desigualdades del Ejemplo 1.4.

en el nodo 1), que es una cota superior de f_1 ó $-f_1$ (el flujo puede ir en cualquier dirección). Argumentos similares explican el significado físico de las restantes condiciones de compatibilidad, que son balances de flujo en diferentes partes de la red suponiendo que circula el flujo máximo en todas sus conducciones.

El Conjunto de todas las Soluciones Factibles

Ahora el sistema (1.19) se resuelve para el caso particular

$$(f_1, f_2, f_3, f_4)^T = (7, -4, -1, -2)^T$$

y $m_{ij} = 4; \;\; \forall i, j$. Utilizando una variable auxiliar $y = 1$ el sistema (1.19) puede escribirse como

$$\begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -19 \\ -1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 4 \\ 0 & -1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 14 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & -8 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & -8 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & -8 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & -8 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & -8 \end{pmatrix} \begin{pmatrix} y_{12} \\ y_{13} \\ y_{14} \\ y_{24} \\ y_{34} \\ z_{12} \\ z_{13} \\ z_{14} \\ z_{24} \\ z_{34} \\ y \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \qquad \text{(A.63)}$$

Tabla A.16: Cono dual asociado al sistema (A.63)–(A.64)

$\mathbf{w}_1$	$\mathbf{w}_2$	$\mathbf{w}_3$	$\mathbf{w}_4$
4	8	4	8
8	8	7	3
7	3	8	8
0	4	0	4
7	7	6	2
4	0	4	0
0	0	1	5
1	5	0	0
8	4	8	4
1	1	2	6
1	1	1	1

y

$$\begin{pmatrix} -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \end{pmatrix} \begin{pmatrix} y_{12} \\ y_{13} \\ y_{14} \\ y_{24} \\ y_{34} \\ z_{12} \\ z_{13} \\ z_{14} \\ z_{24} \\ z_{34} \\ y \end{pmatrix} \leq \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \tag{A.64}$$

y se obtiene el cono dual del cono generado por las filas de las matrices en (A.63)–(A.64), que se da en la tabla A.16. Ahora, forzando la condición $y = 1$ y volviendo a las variables iniciales x se obtiene como solución general el politopo:

$$\begin{array}{rcl} \mathbf{x} & = & \lambda_1(0,4,3,-4,3)^T + \lambda_2(4,4,-1,0,3)^T \\ & & +\lambda_3(0,3,4,-4,2)^T + \lambda_4(4,-1,4,0,-2)^T\} \\ 1 & = & \lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 \end{array} \tag{A.65}$$

Nótese que los cuatro vectores de (A.65) satisfacen el sistema inicial (1.19).

Ejercicios

A.1 Considérese la red de abastecimiento de agua de la Figura A.4, donde $\{q_1, q_2\}$ son los caudales suministrados, $\{q_3, \ldots, q_{10}\}$ son los caudales consumidos en los nodos de la red, y $\{x_1, \ldots, x_{14}\}$ son los caudales correspondientes en las conducciones. Por convención, se considera que $q_1, q_2 < 0$ y $q_i \geq 0$ para $i = 3, \ldots, 14$.

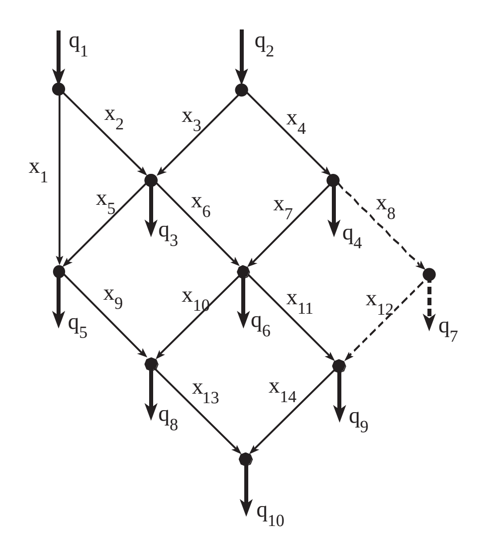


Figura A.4: Sistema de abastecimiento de agua mostrando los caudales en todas las conducciones.

(a) Escribir el balance de caudales en cada nodo para obtener el sistema de ecuaciones de la red de abastecimiento.

(b) Obtener las condiciones de compatibilidad y darles una interpretación física.

(c) Suponer que los caudales disponibles satisfacen la condición $q_1 = q_2$ y que los caudales consumidos son $q_3 = q_4 = q_5 = q_6 = q_7 = q_8 = q_9 = q_{10} = 1 \;\; m^3/segundo$. ¿Cuál el valor común de q_1 y q_2?

(d) Obtener la solución general del sistema de ecuaciones en este caso.

(e) Considérense las dos situaciones siguientes:

(i) El consumidor q_7 tiene suministro, pero su consumo es nulo.

(ii) Se corta el suministro al consumidor q_7.

Obtener las nuevas soluciones a partir de las antiguas.

A.2 Considérese la red de la Figura A.5. Muéstrese que para este caso la

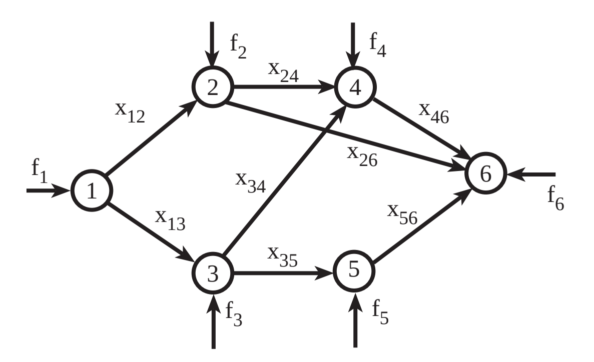


Figura A.5: Un problema de flujo en redes.

ecuación (1.16) se transforma en

$$
\begin{pmatrix}
1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\
0 & -1 & 0 & 0 & 1 & 1 & 0 & 0 \\
0 & 0 & -1 & 0 & -1 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 \\
0 & 0 & 0 & -1 & 0 & 0 & -1 & -1
\end{pmatrix}
\begin{pmatrix} x_{12} \\ x_{13} \\ x_{24} \\ x_{26} \\ x_{34} \\ x_{35} \\ x_{46} \\ x_{56} \end{pmatrix}
=
\begin{pmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \\ f_5 \\ f_6 \end{pmatrix}
$$

$$
\begin{array}{rcl}
x_{ij} & \leq & m_{ij}; \forall i,j \\
-x_{ij} & \leq & m_{ij}; \forall i,j
\end{array}
\tag{A.66}
$$

Definiendo las nuevas variables $y_{ij} = x_{ij} + m_{ij}; \forall i, j$, mostrar que el sistema (A.66) se transforma en

$$
\begin{pmatrix}
1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\
0 & -1 & 0 & 0 & 1 & 1 & 0 & 0 \\
0 & 0 & -1 & 0 & -1 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 \\
0 & 0 & 0 & -1 & 0 & 0 & -1 & -1
\end{pmatrix}
\begin{pmatrix} y_{12} \\ y_{13} \\ y_{24} \\ y_{26} \\ y_{34} \\ y_{35} \\ y_{46} \\ y_{56} \end{pmatrix}
=
\begin{pmatrix} f_1^* \\ f_2^* \\ f_3^* \\ f_4^* \\ f_5^* \\ f_6^* \end{pmatrix}
$$

$$
\begin{array}{rcl}
y_{ij} + z_{ij} & = & 2m_{ij}; \forall i,j \\
y_{ij} & \geq & 0; \forall i,j \\
z_{ij} & \geq & 0; \forall i,j
\end{array}
\tag{A.67}
$$

donde $z_{ij} \geq 0; \forall i, j$ son las variables de holgura utilizadas para convertir

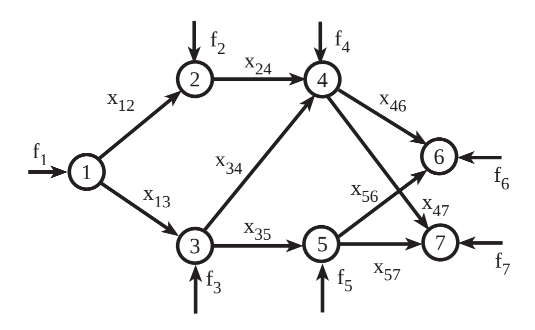


Figura A.6: Problema de flujo en redes.

el sistema de inecuaciones en el sistema de ecuaciones (A.67), y

$$\begin{pmatrix} f_1^* \\ f_2^* \\ f_3^* \\ f_4^* \\ f_5^* \\ f_6^* \end{pmatrix} = \begin{pmatrix} f_1 - m_{12} - m_{13} \\ f_2 + m_{12} - m_{24} - m_{26} \\ f_3 + m_{13} - m_{34} - m_{35} \\ f_4 + m_{24} + m_{34} - m_{46} \\ f_5 + m_{35} - m_{56} \\ f_6 + m_{26} + m_{46} + m_{56} \end{pmatrix} \tag{A.68}$$

Obtener las condiciones de compatibilidad y la solución general del problema.

A.3 Considérese la red de flujo de la Figura A.6.

(a) Escribir el sistema de ecuaciones correspondiente.

(b) Obtener las condiciones de compatibilidad y darles un significado físico.

(c) Obtener el conjunto de todas las soluciones factibles y explicar el resultado.

(d) Obtener la solución óptima para costes unidad y $f_i = 10; \forall i$.

(e) Suponer que falla el envío asociado a x_{47}; rehacer las partes (a)–(d) sin comenzar de nuevo.

A.4 En el problema siguiente de planificación de la producción, considérese la demanda de la Figura A.7 (véase también la tabla A.17) y supóngase una disponibilidad inicial de dos unidades.

(a) Escribir el sistema de ecuaciones asociadas.

(b) Obtener las condiciones de compatibilidad y darles una interpretación física.

(c) Obtener el conjunto de soluciones factibles y explicar el resultado.

(d) Fijar los costes requeridos.

(e) Obtener la solución óptima para estos costes.

Tabla A.17: Tabla de demanda

Tiempo	1	2	3	4	5	6	7	8	9	10	11	12
Demanda	3	4	5	4	3	2	3	1	2	1	3	4

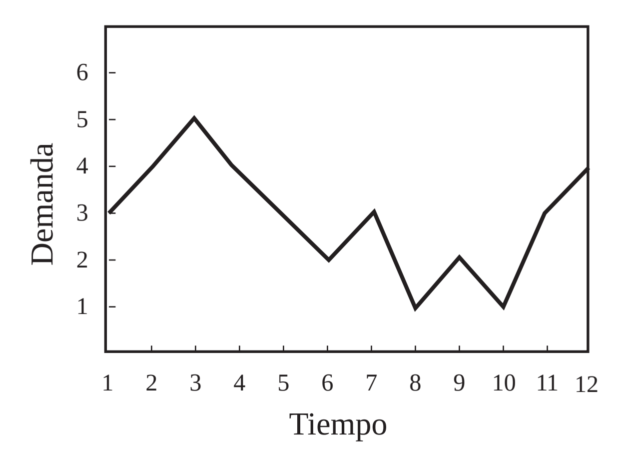


Figura A.7: Problema de la planificación de la producción.

Apéndice B

Notación

En este apéndice se lista la notación principal usada en este libro. Se ha tratado de que la notación sea todo lo consistente posible a lo largo del libro.

Capítulo 1

a_{ij}	Cantidad de nutriente i por unidad de alimento j
a_t	Precio de venta de una unidad de alimento en t
b_i	Número actual de participaciones del valor i
	Mínima cantidad requerida de nutriente i
b_t	Coste de producción en el instante t
B	Conjunto de vigas
B_{ij}	Susceptancia de la línea i–j
c_j	Coste de la unidad de alimento j
c_t	Coste de almacenamiento en el instante t
c_{ij}	Coste de enviar una unidad de producto del origen i al destino j
C_i	Coste de producir energía eléctrica en el generador i
d_i	Dividendo que se paga al final del año por la acción i
d_t	Demanda de unidades en el mes t
dl_i	Distancia de la carga i al extremo izquierdo de la viga en la que actúa
dr_s	Distancia de la cuerda s al extremo izquierdo de la viga b que soporta
D_i	Demanda en el nudo i
f_i	Flujo, de entrada en el nodo i
$\mathcal{G}$	Grafo $\mathcal{G} = (\mathcal{N}, \mathcal{A})$ donde $\mathcal{N}$ es el conjunto de nodos y $\mathcal{A}$ es el conjunto de arcos
I	Conjunto de cargas
m	Número de orígenes
	Número de nutrientes

	Número de acciones
m_{ij}	Máxima capacidad del arco que va del nodo i al nodo j
n	Número de alimentos diferentes
	Número de meses a considerar
	Número de generadores de energía eléctrica
	Número de destinos
	Número de nodos en la red
$\underline{P}_i$	Mínima potencia del generador i
p_i	Potencia activa producida por el generador i
$\overline{P}_i$	Máxima potencia del generador i
P_{ij}^{max}	Capacidad máxima de transmisión de la línea i–j
s_0	Cantidad de producto disponible en el almacén al comienzo del periodo considerado
s_t	Número de unidades almacenadas en el mes t
s_t^{max}	Capacidad de almacenamiento
S	Conjunto de cuerdas
t_s	Tensión generada en la cuerda s bajo la acción del conjunto de cargas
T_s	Máxima carga admisible en la cuerda s
u_i	Cantidad de producto a mandar desde el origen i
v_i	Valor actual de la acción i
v_j	Cantidad de producto a recibir en el destino j
w_i	Nuevo valor de la acción i
x_i	Carga i
	Cambio en el número de acciones de la empresa i
x_{ij}	Cantidad a enviar desde el origen i al destino j
	Flujo que va del nodo i al nodo j
x_j	Cantidad a comprar de alimento j
x_t	Número de unidades producidas en el mes t
$\mathbf{z}^T$	Transpuesta del vector $\mathbf{z}$
Z	Valor de la función objetivo
δ_i	Ángulo del nudo i
Ω_b	Conjunto de cargas aplicadas en la viga b
Ω_i	Conjunto de nudos conectados al nudo i
Ψ_b	Conjunto de cuerdas que soportan la viga b
Θ_b	Conjunto de cuerdas soportadas por la viga b

Capítulo 2

a	Nivel de discrepancia mínimo requerido
a_j	Peso del objeto j en el problema de la mochila
A_j	Coste fijo de la unidad térmica j
b	Capacidad de la mochila
b_i	Demanda del cliente i-ésimo
B_j	Coste variable de la unidad térmica j

c_{ij}	Nivel del síntoma j asociado a la enfermedad i
	Beneficio unitario de los productos originados en la localización j y vendidos al cliente i
c_j	Utilidad del objeto j
C_j	Puntuación total del candidato j
	Coste de arranque de la central térmica j
$d(x)$	Una medida de discrepancia
$\mathcal{D}$	Conjunto de enfermedades
D_k	Demanda en el periodo k
E_j	Coste de parada de la central térmica j
f_j	Coste fijo de apertura de la fábrica localizada en j
I	Número actual de miembros de la Academia de Ingeniería
	Conjunto $\{1, \ldots, n\}$ de n clientes
J	Número de candidatos
	Número de unidades de potencia
	Conjunto $\{1, \ldots, m\}$ de los m lugares donde se pueden situar las fábricas
K	Número de unidades de tiempo
m	Número de síntomas
n	Número de enfermedades
	Número de objetos del problema de la mochila
n_b	Número de bloques académicos
n_c	Número de aulas
n_h	Número de horas de docencia disponibles
n_i	Número de asignaturas que enseña el profesor i
n_s	Número de asignaturas
p_{jk}	Potencia de salida de la central térmica j durante el periodo k
$\underline{P}_j$	Mínima potencia de salida de la unidad j
p_s	la puntuación s-ésima
$\overline{P}_j$	Máxima potencia de salida de la central térmica j
P_j^0	Potencia de salida de la central térmica j justo antes del primer periodo del horizonte de planificación
r	Número de nuevos miembros a incorporar
R_k	Cantidad de reserva requerida en el periodo k
S	Número de puntuaciones diferentes a asignar
S_j	Máximo incremento de potencia en rampa de la central térmica j
$\mathcal{S}$	Conjunto de síntomas
$\mathcal{S}_a$	Un subconjunto del conjunto de síntomas $\mathcal{S}$
T_j	Máximo decremento de potencia en rampa de la central térmica j
u_j	Capacidad de la fábrica localizada en j
$v(s, c, h)$	Variable binaria que toma el valor 1 si la asignatura s se imparte en el aula c a la hora h, y 0, en otro caso

v_{jk}	Variable binaria que es igual a 1, si la central térmica j está conectada a la línea durante el periodo k y 0, en otro caso
V_j^0	Constante binaria que es igual a 1, si la central térmica j está conectada a la línea durante el periodo anterior al primero del horizonte de planificación, y 0, en otro caso
x_{ij}	Cantidad de producto enviada desde la fábrica j al cliente i
x_{ijs}	Variable binaria igual a 1 si el miembro i asigna la puntuación p_s al candidato j, y 0, en otro caso
y_j	Variable binaria para modelar la elección de abrir una fábrica en el lugar j
y_{jk}	Variable binaria igual a 1 si la central térmica j se arranca al comienzo del periodo k y 0, en otro caso
z_{jk}	Variable binaria igual a 1 si la central térmica j se para al comienzo del periodo k y 0, en otro caso
Z	Valor de la función objetivo
Δ_b	Conjunto de asignaturas del bloque académico b
Ω	Conjunto de todas las asignaturas a enseñar
Ω_i	Conjunto de asignaturas que enseña el profesor i

Capítulo 3

a	Dimensión de la base cuadrada de la tienda
$(\mathcal{A},\mathcal{N})$	Un grafo dirigido $(\mathcal{A},\mathcal{N})$ donde $\mathcal{A}$ es el conjunto de arcos y $\mathcal{N}$ es el conjunto de nodos
b	Altura de las paredes verticales de la tienda
B	Conjunto de vigas
c_j	Número de viajes atraídos por la zona j
	Suma de los elementos de la columna j de la matriz
$C_a(f_a)$	Función de coste asociada al arco a
C_i	Coste de producir una unidad de potencia activa usando el generador i
d_ω	Número de viajes en coche desde el origen i al destino j, para cualquier par origen-destino ω
D	Desplazamiento
	Peso específico del material a usar
D_0	Máximo desplazamiento admisible
$D(x, y, z)$	Función distancia
E	Módulo de Young del material a utilizar
f_a	Flujo en el arco a
F	Carga en la parte superior de la ménsula
	Carga aplicada
g	Aceleración de la gravedad
h	Altura de la pirámide

	Altura de la lámpara
	Altura de la estructura
h_r	Flujo en la ruta r
H	Altura total de la tienda
	Altura de la columna
I	Intensidad de la luz
	Momento de inercia
l_B	Longitud total de la viga b
L	Longitud de la ménsula
k	Constante de proporcionalidad
M	Masa a soportar por la columna
n	Número de nudos en la red eléctrica
$\hat{p}_{kl}$	Medida de la potencia activa saliendo del nudo k hacia el nudo l de la línea k–l
p_{Gi}	Generación de potencia activa en el nudo i
P_{Di}	Demanda de potencia activa en el nudo i
$\underline{P}_{Gi}$	Mínima potencia activa del generador i
$\overline{P}_{Gi}$	Máxima potencia activa del generador i
$\hat{q}_{kl}$	Medida de la potencia reactiva saliendo del nudo k hacia el nudo I de la línea k–l
q_{Gi}	Generación de potencia reactiva en el nudo i
$\underline{Q}_{Gi}$	Mínima potencia reactiva del generador i
$\overline{Q}_{Gi}$	Máxima potencia reactiva del generador i
Q_{Di}	Demanda de potencia reactiva en el nudo i
r_i	Número de viajes generados en la zona i
	Suma de la fila i de la matriz
$\mathcal{R}_\omega$	Conjunto de rutas simples para el par de demanda ω
S	Máxima deformación posible de la ménsula
	Superficie de la tienda
	Máxima tensión actual (fuerza por unidad de superficie)
S_0	Máxima tensión admisible
S^i	Tensión en el punto i
t_{ij}	Contenido observado en la celda i–j
	Viajes observados desde i a j
T_{ij}	Predicción de viajes desde i a j
v_i	Magnitud de la tensión en el nudo i
$\hat{v}_i$	Magnitud medida de la tensión en el nudo i
V	Volumen total de la tienda
	Volumen total del paquete postal
$\underline{V}_i$	Cota inferior de la magnitud de la tensión en el nudo i
$\overline{V}_i$	Cota superior de la magnitud de la tensión en el nudo i
W	Peso total de la estructura de barras
	Conjunto de pares origen–destino
x	Distancia de los puntos fijos al eje Y
xl_i	Distancia de la carga i el extremo izquierdo de la viga

	donde actúa
(y_{ik}, θ_{ik})	Una constante compleja dada por su módulo y_{ik}, y argumento θ_{ik}
z	Área de la sección transversal de las barras
$z_{k\ell}$	Magnitud de la impedancia asociada a la línea k–l
δ_i	Ángulo de la tensión en el nudo i
γ or ρ	Peso específico del material de la barra
Ω	Conjunto de nudos de la red eléctrica
Ω_k	Conjunto de nudos conectados al nudo k
σ_i^v	Desviación típica de la tensión medido
σ_{kl}^p	Desviación típica de $\hat{p}_{kl}$
σ_{kl}^q	Desviación típica de $\hat{q}_{kl}$
$\theta_{k\ell}$	Ángulo de la impedancia asociada a la línea k–l

Capítulo 4

$\mathbf{A}$	Matriz asociada a las restricciones en un PPL
$\mathbf{A} = (\mathbf{B} \quad \mathbf{N})$	Descomposición en columnas básicas y no básicas
$\mathbf{b}$	Vector que define las restricciones en un PPL
$\mathbf{B}$	Una matriz básica factible
$\mathbf{B}^*$	Matriz básica óptima
$\mathbf{c}$	Vector que define la función objetivo en un PPL
$\mathbf{c}_B^T$	Vector básico de coste
D	Problema dual
P	Problema primal
x^+, x^-	Partes positiva y negativa de un número
$\mathbf{x}_B$	Vector de todas las variables básicas
$\mathbf{x}_B^*$	Solución básica óptima
$\tilde{\mathbf{x}}$	Un vector factible para el problema primal
$\mathbf{y}$	Vector de las variables duales
$\tilde{\mathbf{y}}$	Un vector factible para el problema dual
z^*	Valor óptimo de la función objetivo
$Z = f(\mathbf{x})$	Función objetivo o de coste
λ_k	Coeficientes de una combinación lineal convexa
$\boldsymbol{\lambda}^{*T}$	Vector de los parámetros de sensibilidad o de las variables duales
π_j	Coeficientes de una combinación lineal no negativa
ρ_i	Coeficientes de una combinación lineal de vectores
$\Delta\mathbf{b}$	Cambio marginal en el vector $\mathbf{b}$
$\Delta\mathbf{x}_B$	Cambio en la solución óptima
Δz	Cambio en el valor óptimo de la función objetivo

Capítulo 5

$\mathbf{a}_k$	Vector columna k-ésimo de la matriz $\mathbf{A}$
$\mathbf{A}$	Una matriz típica

$\mathbf{A}_\lambda$	Conjunto de todas las combinaciones lineales convexas de los vectores columna de **A**
$\mathbf{A}_\pi$	Conjunto de todas las combinaciones lineales no negativas de los vectores columna de **A**
$\mathbf{A}_\pi \equiv \mathbf{B}_\rho + \mathbf{C}_\pi$	Forma general de un cono como la suma de un espacio vectorial y un cono propio
$\mathbf{A}_\rho$	Conjunto de todas las combinaciones lineales de los vectores columna de **A**
$\mathbf{B}$	Conjunto de generadores cuyos opuestos son del cono
C	Un cono convexo genérico
$\mathbf{C}$	Conjunto de generadores cuyos opuestos no pertenecen al cono
$\mathbf{d}$	Una dirección arbitraria de un conjunto convexo
D	El conjunto de las direcciones
D_E	El conjunto de las direcciones extremas
E	Conjunto de puntos extremos de un conjunto convexo
$\mathbf{H}$	Matriz que define un espacio vectorial, un cono, un politopo, o un poliedro
$\mathbf{p}$	Un vector no nulo de $\mathbb{R}^n$
$\mathbf{q}_k$	Generador k-ésimo de un politopo
$\mathbf{Q}$	Conjunto de puntos extremos
S	Un conjunto convexo arbitrario
S^+, S^-	Semiespacios asociados a S
$S_1 + S_2$	La suma de dos conjuntos convexos
$\{S_i\}_{i\in I}$	Una familia de conjuntos convexos
$\mathbf{v}_i$	Generador i-ésimo de un espacio vectorial
$\mathbf{V}_\rho + \mathbf{W}_\pi + \mathbf{Q}_\lambda$	Forma general de un poliedro
$\mathbf{w}_j$	Generador j-ésimo de un cono
$\mathbf{W}$	Conjunto de direcciones extremas
α	Un escalar arbitrario
λ_k	Coeficientes de las combinaciones lineales convexas
$\lambda\mathbf{x} + (1-\lambda)\mathbf{y}$	Una combinación lineal convexa típica
ρ_i	Coeficientes de las combinaciones lineales
π_j	Coeficientes de las combinaciones lineales no negativas
$\mathcal{V}$	Un espacio vectorial
$\mathcal{W}$	Un subespacio vectorial

Capítulo 6

$\mathbf{A}$	Matriz $m \times n$ de los coeficientes de las restricciones
$(\mathbf{B} \quad \mathbf{N})$	Partición de **A** en columnas básicas y no básicas
$\mathbf{c} = (c_1, c_2, \ldots, c_n)^T$	Matriz columna de los coeficientes de coste
$(\mathbf{c}_\mathbf{B}^T \quad \mathbf{c}_\mathbf{N}^T)$	Partición de los coeficientes de la función objetivo en sus componentes básicas y no básicas
(t)	Número de la iteración
$\mathbf{x} = (x_1, x_2, \ldots, x_n)^T$	Vector columna de las variables iniciales

$(\mathbf{x_B} \quad \mathbf{x_N})$	Partición de $\mathbf{x}$ en componentes básicas y no básicas
$z_{\alpha\beta}^{(t)}$	Un pivote
Z	Valor de la función objetivo
$\mathbf{Z}^{(t)}$	Matriz de las iteraciones del método MS a las que se aplica la operación elemental de pivotación
α	Fila del pivote
β	Columna del pivote
$\Lambda_\alpha^{(t)}$	Razón función-pivote

Capítulo 7

$(\mathbf{B} \quad \| \quad \mathbf{N})$	Partición de la matriz $\mathbf{A}$ en sus componentes básicas $\mathbf{B}$ y no básicas $\mathbf{N}$
$\mathbf{I}$	Matriz identidad
$\mathbf{U} = \mathbf{B}^{-1}\mathbf{N}$	Matriz central en el algoritmo de Gomory
$\mathbf{x}_B \cup \mathbf{x}_N$	Partición del conjunto de variables $\{x_1, x_2, \dots, x_n\}$ en básicas $\mathbf{x}_B$ y no básicas $\mathbf{x}_N$
x_{B_i}	Variables básicas no enteras a convertirse en enteras
i_{ij}	Parte entera de u_{ij}
f_{ij}	Parte fraccional de u_{ij}
Z	Función objetivo lineal

Capítulo 8

$\mathbf{A}^{-1}$	Inversa de la matriz $\mathbf{A}$
$c_{ti}(p_{ti})$	Coste de producción del productor i en el periodo t para producir p_{ti}
d	Un dirección cualquiera
d_t	Demanda en el periodo t
$\det(\mathbf{A})$	Determinante de la matriz $\mathbf{A}$
$f(\mathbf{x})$	Función objetivo $f : \mathbb{R}^n \to \mathbb{R}$
$\mathbf{F}(z)$	Condición de KKT para un problema de restricciones de igualdad
$\mathbf{g}(\mathbf{x})$	Restricción de desigualdad $\mathbf{g} : \mathbb{R}^n \to \mathbb{R}^m$
$\mathbf{h}(\mathbf{x})$	Restricción de igualdad $\mathbf{h} : \mathbb{R}^n \to \mathbb{R}^\ell$
I	Número de productores
$I(\bar{\mathbf{x}})$	Conjunto de índices activo en $\mathbf{x}$ para las restricciones de desigualdad $I(\bar{\mathbf{x}}) = \{j \| g_j(\bar{\mathbf{x}}) = 0\}$
$\text{Inf}_{\ \mathbf{x}\in S}\ q(\mathbf{x})$	Ínfimo de la función q en S
$\text{Inf}_{\ \mathbf{x}}\ q(\mathbf{x}, \mathbf{y})$	Ínfimo de la función q con respecto a la variable $\mathbf{x}$
$\mathcal{L}(\mathbf{x}, \boldsymbol{\mu}, \boldsymbol{\lambda})$	Función Lagrangiana para el PPN
$\lim_{\mathbf{y}\to\mathbf{x}} q(\mathbf{y})$	Límite de la función q en $\mathbf{x}$
$\max_{\mathbf{x}\in S} q(\mathbf{x})$	Máximo de la función q en S
$\min_{\mathbf{x}\in S} q(\mathbf{x})$	Mínimo de la función q en S
p_{ti}	Cantidad producida por el productor i en el periodo t

$\mathbb{R}$	Conjunto de los números reales
$\mathbb{R}^n$	Espacio real n-dimensional
S	Conjunto de las soluciones factibles
$\text{Sup}_{\mathbf{x}}\ q(\mathbf{x},\mathbf{y})$	Supremo de la función q con respecto a la variable $\mathbf{x}$
T	Número de intervalos de tiempo
$\mathbf{x}$	Vector de las variables de decisión
x_i	Coordenada i del vector de las variables de decisión
$\bar{\mathbf{x}}$	Mínimo local de un problema de optimización
$\mathbf{x}^*$	Mínimo global de un problema de optimización
$\mathbf{x}(\boldsymbol{\lambda},\boldsymbol{\mu})$	Mínimo del Lagrangiano para los multiplicadores $(\boldsymbol{\lambda},\boldsymbol{\mu})$
$\mathbf{z}^T$	Transpuesta del vector $\mathbf{z}$
$\mathbf{z}^T\mathbf{y}$	Producto escalar estándar de los vectores $\mathbf{z}$ e $\mathbf{y}$ de $\mathbb{R}^n$
Z	Valor de la función objetivo
$\boldsymbol{\Delta}\mathbf{z}$	Incremento de la variable $\mathbf{z}$
$\lambda_t^{(k)}$	Precio de venta para el periodo t
$\boldsymbol{\lambda}$	Vector de los multiplicadores de Lagrange asociados a las restricciones de igualdad
$\boldsymbol{\mu}$	Vector de los multiplicadores de Lagrange asociados a las restricciones de desigualdad
$(\boldsymbol{\lambda},\boldsymbol{\mu})$	Variables duales
$\sigma(t)$	Una curva en $\mathbb{R}^n$ $(\sigma : (-\epsilon,\epsilon) \to \mathbb{R}^n)$
$\theta(\boldsymbol{\lambda},\boldsymbol{\mu})$	Función dual
Π_i	Región factible de operación del productor i
$\dfrac{\partial q}{\partial x_i}$	Derivada parcial de q con respecto a x_i
$q'(x)$	Primera derivada de la función q
∇q	Gradiente de la función q
	$m \times n$ Jacobiano de $q : \mathbb{R}^n \to \mathbb{R}^m$ $(m \geq 2)$
$\nabla_{\mathbf{y}} q(\mathbf{x},\mathbf{y})$	Matriz Jacobiana de q con respecto a $\mathbf{y}$
$\nabla^2 q$	Matriz hessiana de q
$\nabla_{\mathbf{yy}} q(\mathbf{x},\mathbf{y})$	Matriz hessiana de q con respecto a $\mathbf{y}$
$\|\cdot\|$	Norma euclídea de un vector
[a,b]	Intervalo cerrado

Capítulo 9

arg minimizar$_{\mathbf{x}} q(\mathbf{x})$	Valor de $\mathbf{x}$ que conduce al mínimo de la función q
$\mathbf{A}^{-1}$	Inversa de la matriz $\mathbf{A}$
$\mathbf{B}^{(t)}$	Aproximación a la matriz hessiana $[\nabla^2 f(\mathbf{x}^{(t)})]$
D	Conjunto de las soluciones factibles del problema dual
$\mathbf{d}$	Una dirección de búsqueda
$\mathbf{d}^{(t)}$	Dirección de búsqueda en la iteración t
$f(\mathbf{x})$	Función objetivo
$\mathbf{g}(\mathbf{x})$	Restricción de desigualdad $\mathbf{g} : \mathbb{R}^n \to \mathbb{R}^m$
$\mathbf{h}(\mathbf{x})$	Restricción de igualdad $\mathbf{h} : \mathbb{R}^n \to \mathbb{R}^\ell$

$\mathbf{H}^{(t)}$	Aproximación a la inversa de la matriz hessiana de f en $\mathbf{x}^{(t)}$
Ínfimo $_{\mathbf{x}\in S}\ q(\mathbf{x})$	Ínfimo de la función q en S
Inf $_{\mathbf{x}}\ q(\mathbf{x},\mathbf{y})$	Ínfimo de la función q con respecto a la variable $\mathbf{x}$
ℓ	Cota inferior
$\mathcal{L}(\mathbf{x},\boldsymbol{\mu},\boldsymbol{\lambda})$	Función Lagrangiana
$\lim_{t\to+\infty} x_t$ or x_∞	Límite de la sucesión de números reales $\{x_t\}$
$\lim_{\mathbf{y}\to\mathbf{x}} q(\mathbf{y})$	Límite de la función q en $\mathbf{x}$
$\max_{\mathbf{x}\in S} q(\mathbf{x})$	Máximo de la función q en S
$\min_{\mathbf{x}\in S} q(\mathbf{x})$	Mínimo de la función q en S
$\mathbf{P}$	Matriz de proyección
$P(\mathbf{x};r)$	Una función genérica de penalización exterior
$P_1(\mathbf{x};r)$	Función de penalización valor absoluto
$P_2(\mathbf{x};r)$	Función de penalización cuadrática
r	Parámetro de penalización
r_d	Residuo dual para el método del punto interior
r_p	Residuo primal para el método del punto interior
r_t	Parámetro de penalización en la iteración t
$\mathbb{R}$	Conjunto de los números reales
$\mathbb{R}^n$	Espacio real n-dimensional
S	Conjunto de las soluciones factibles del primal
S^-	Un subconjunto de la región factible S definida por $S^- = \{\mathbf{x}\mid\mathbf{g}(\mathbf{x}) < \mathbf{0}\}$
Sup $_{\mathbf{x}}\ q(\mathbf{x},\mathbf{y})$	Supremo de la función q con respecto a la variable $\mathbf{x}$
t	Contador de iteraciones
u	Cota superior
$\mathbf{x}$	Vector de las variables primales
$\mathbf{x}^{(t)}$	$\mathbf{x}$-vector de las iteraciones
x_t	Escalar
$\{\mathbf{x}^{(t)}\}$	Sucesión de vectores
$\{x_t\}$	Sucesión de escalares
$\mathbf{x}(r)$	Mínimo de la función de penalización $P(\mathbf{x},r)$
$\mathbf{x}(\boldsymbol{\lambda},\boldsymbol{\mu})$	Mínimo de la función Lagrangiana $(\boldsymbol{\lambda},\boldsymbol{\mu})$
$X(\boldsymbol{\lambda},\boldsymbol{\mu})$	Conjunto de puntos $\mathbf{x}(\boldsymbol{\lambda},\boldsymbol{\mu})$
$\mathbf{z}^T$	Traspuesto del vector $\mathbf{z}$
$\mathbf{z}^T\mathbf{y}$	Producto escalar estándar de vectores en $\mathbb{R}^n$
$Z,\ Z_P$	Valor de la función objetivo del problema primal
Z_D	Valor de la función objetivo del problema dual
Z_D^*	Valor óptimo del problema dual
Z_P^*	Valor óptimo del problema primal
Z_{LS}	Valor de la función objetivo en la búsqueda lineal
$\boldsymbol{\Delta}\mathbf{z}$	Incremento de la variable $\mathbf{z}$
$\boldsymbol{\alpha}$	Longitud de paso
$\boldsymbol{\alpha}_d$	Longitud de paso en el método del punto interior
$\boldsymbol{\alpha}_p$	Longitud de paso primal en el método del punto interior

$\boldsymbol{\alpha}_t$	Longitud de paso en la búsqueda lineal t
β	Un parámetro del método del gradiente conjugado
δ	Parámetro usado en la regla de Armijo Tolerancia en el método del punto interior
$\boldsymbol{\lambda}$	Vector de los multiplicadores de Lagrange asociados a las restricciones de igualdad
$\boldsymbol{\lambda}^c$	Multiplicadores de Lagrange asociados a las columnas de la matriz equilibrada
$\boldsymbol{\lambda}^r$	Multiplicadores de Lagrange asociados a las filas de la matriz equilibrada
$\boldsymbol{\mu}$	Vector de los multiplicadores de Lagrange asociados a las restricciones de desigualdad
η	Parámetro usado para aumentar el valor del parámetro de penalización del método de penalización exterior
ε	Tolerancia para controlar la convergencia
$(\boldsymbol{\lambda}, \boldsymbol{\mu})$	Variables duales
$\psi(\mathbf{z}, \mathbf{y})$	Función de penalización exterior definida como $\psi : \mathbb{R}^{\ell+m} \to \mathbb{R}$
$\phi(\mathbf{x})$	Función de penalización interior
$\theta(\boldsymbol{\lambda}, \boldsymbol{\mu})$	Función dual
$q'(x)$	Primera derivada de la función $q : \mathbb{R} \to \mathbb{R}$
$q'(\mathbf{x}; \mathbf{d})$	Derivada direccional de q en la dirección $\mathbf{d}$ en el punto $\mathbf{x}$
$q''(x)$	Segunda derivada de la función $q : \mathbb{R} \to \mathbb{R}$
∇q	Gradiente de $q : \mathbb{R}^n \to \mathbb{R}$ $m \times n$ Jacobiano de $q : \mathbb{R}^n \to \mathbb{R}^m$ $(m \geq 2)$
$\nabla_{\mathbf{y}} q(\mathbf{x}, \mathbf{y})$	Matriz jacobiana de q con respecto a $\mathbf{y}$
$\dfrac{\partial q}{\partial x_i}$	Derivada parcial de la función q con respecto a x_i
$\left.\dfrac{\partial q}{\partial x_i}\right\|_{\mathbf{x}}$	Derivada parcial de q con respecto a x_i en $\mathbf{x}$
$\nabla^2 q$	Matriz hessiana de $q : \mathbb{R}^n \to \mathbb{R}$
$\\|\cdot\\|$	Norma euclídea de un vector

Capítulo 12

$\overline{a}$	Media del área a
a	Índice que se refiere al transporte en coche
$a(i)$	Área del triángulo i
a_{ij}, b_{ij}	Componentes ij-ésimas de las matrices $\mathbf{A}$, $\mathbf{B}$
a_j	Velocidad del proyectil en el instante j
A	Constante
$\mathbf{A}$	Matriz de probabilidad condicional
$\mathcal{A}$	Conjunto ordenado de los enlaces dirigidos o arcos
A_i	Factor de proporcionalidad de la función $p(c_{ij})$
$A(i)$	Conjunto de nodos “posteriores” al nodo i

b	Transporte realizado en medio público
b_j	Posición del proyectil en el instante j
B	Constante
$\mathbf{B}$	Matriz de probabilidad condicional
$\mathcal{B}$	Subconjunto de los arcos acotados
B_i	Factor de proporcionalidad de la función $p(c_{ij})$
$B(i)$	Conjunto de nodos "anteriores" al nodo i
$c_a(f_a)$	Coste del arco a como función de su nivel de servicio
$\hat{c}_a(f_a)$	Coste del arco a para su capacidad máxima
c_Ω	Coste mínimo de viaje para el par Ω
c_Ω^a	Coste (variable) del viaje en coche asociado al par Ω
c_Ω^b	Coste (constante) del viaje en transporte público asociado al par Ω
c_i^Ω	Coste de la ruta i para un producto Ω
$c_{ij}(x)$	Función de coste del arco en problema de transporte
c_{ij}^0	Tiempo de viaje en el caso de flujo libre
$\mathbf{C}_\pi^*$	Cono dual o polar de $\mathbf{C}_\pi$
$C_{r'}^*$	Longitud de la ruta r'
CPU	Unidad central de proceso
$d_i(LAV)$	Influencia de la observación i-ésima en estimadores MVA
$d_i(LS)$	Influencia de la observación i-ésima en estimadores MC
$d_i(MM)$	Influencia de la observación i-ésima en estimadores MM
$d(\boldsymbol{\lambda}, \boldsymbol{\mu})$	Solución del problema primal dados los valores de $\boldsymbol{\lambda}$, $\boldsymbol{\mu}$ (algoritmo de relajación Lagrangiana)
D	Conjunto de datos de entrada y salida de una neurona Dimensión del espacio de la malla
D_j	Número de viajes atraídos al destino j
$e_{i(jk)}$	Elemento de la fila asociada al nodo i, y la columna asociada al arco (j,k)
$\mathbf{E}$	Matriz de incidencias
$\mathbf{f}$	Flujo de todos los productos
f_{ij}	Flujo en el arco dirigido (i,j)
f_{ij}^Ω	Flujo en el arco dirigido (i,j) asociado al producto Ω
$f(\mathbf{x})$	Función objetivo primal del problema CHTCP
g_Ω	Demanda (número de viajes) del producto Ω
$\overline{g}_\Omega$	Suma de las demandas del producto Ω considerando todos los posibles modos de transporte
g^Ω	Flujo del producto Ω en O^Ω–D^Ω
$\mathbf{g}_i(\mathbf{x}_i)$	Contribución de la central térmica i para satisfacer la reserva de potencia
$\mathbf{g}_j(\mathbf{x}_j)$	Contribución del sistema hidroeléctrico j para satisfacer la reserva de potencia
g_{ij}^*, c_{ij}^*	Valores óptimos de las funciones g_{ij}, c_{ij}, respectivamente
$\mathbf{G}$	Vector de reserva de potencia del problema CHTCP

$\mathcal{G}$	Grafo dirigido
$G_\Omega(c_\Omega)$	Variable demanda como función de c_Ω
$\mathbf{h}$	Vector de flujo
h_r	Flujo en la ruta r
$\mathbf{h}_i(\mathbf{x}_i)$	Contribución de la central térmica i para satisfacer la demanda
$\mathbf{h}_j(\mathbf{x}_j)$	Contribución del sistema hidroeléctrico j para satisfacer la demanda
H	Distancia entre los dos extremos del cable
$\mathbf{H}$	Vector demanda del problema CHTCP
i	Índice de los elementos de la malla
(i,j), a	Elementos del conjunto $\mathcal{A}$
I	Dimensión del vector de datos de entrada
	Número de centrales térmicas
$I(u)$	Funcional de coste
k	Constante
k_a	Máxima capacidad del arco a
$\mathcal{K}$, k	Conjunto de arcos, nodos, rutas, o cualquier combinación de ellos
l	Número de rutas utilizadas
L	Longitud del cable colgante
$\mathcal{L}(.)$	Función Lagrangiana
LAV	Método del menor valor absoluto
LR	Relajación Lagrangiana
m	Número de rutas saturadas
MM	Método Minimax
n	Número de intervalos de discretización
	Número de nodos de la malla
N	Dominio de la variable (X,Y)
$\|N\|$	Cardinalidad del conjunto N
$\mathcal{N}$	Conjunto finito desordenado de los nodos del grafo $\mathcal{G}$
O-D	Par origen-destino
O_i	Número de viajes que parten del origen i
$p_{i.}$	Probabilidad marginal de X
$p_{.j}$	Probabilidad marginal de Y
p_{ij}	Probabilidad conjunta de (X,Y)
$p(c_{ij})$	Función de disuasión
$\mathbf{P}$	Matriz de probabilidad conjunta
PP	Problema primal
Q	Valor de la función objetivo
Q_j	Valor de la función objetivo asociada a la opción de aprendizaje j
r, r'	Índices de las rutas
r_i	Flujo producido o atraído por el nodo i
r_i^Ω	Flujo variable producido o atraído por el nodo i

r	Índice de las funciones neuronales básicas invertibles
R	Número de funciones neuronales básicas invertibles
$\mathcal{R}_\Omega$	Conjunto de rutas para el producto Ω
$\mathbf{s}^{(t)}$	Subgradiante de la función dual en la iteración t
$s_k(\mathbf{f})$	Función genérica k-ésima dependiente del flujo en los arcos
S	Número de medidas en la red neuronal
STHTC	Coordinación hidrotérmica a corto plazo
t	Tiempo
(t)	Índice de la iteración
$u(t)$	Función que establece el control para modificar la dinámica de un sistema dado algún objetivo específico
$u'(t)$, $u''(t)$	Primera y segunda derivadas de la función $u(t)$
u_a	Máxima capacidad del arco a
$\mathbf{V}$, $\mathbf{W}$	Matrices de generadores del cono dual $\mathbf{C}_\pi^*$
W	Conjunto de todos los pares O-D
w_i	Peso correspondiente a la entrada i de la neurona
w_0	Valor umbral de la neurona
$\mathbf{x}_i$	Vector de variables primales asociadas a la central térmica i
$\mathbf{x}_j$	Vector de variables primales asociadas al sistema hidráulico j
x_{is}	Vector de datos de una neurona
x_{max}	Máxima coordenada x de la malla
x_{min}	Mínima coordenada x de la malla
$\mathbf{x}_i^{*(t)}$	Valor óptimo del vector $\mathbf{x}_i$ en la iteración t
$x(t)$	Función que determina la dinámica del estado del sistema bajo control
(X, Y)	Variable bidimensional discreta
$\mathbf{y}$	Variable respuesta en el modelo de regresión
$\hat{\mathbf{y}}$	Vector de las predicciones de la variable respuesta $\mathbf{y}$ con todos los datos
$\hat{\mathbf{y}}_{(i)}$	Vector de las predicciones de la variable respuesta $\mathbf{y}$ cuando se omite la observación i
y_i	Componente i-ésima del vector $\mathbf{y}$
y_{max}	Máxima valor de la coordenada y en la malla
y_{min}	Mínimo valor de la coordenada y en la malla
y_s	Vector de salida de la neurona
y_{s_1}, y_{s_2}	Componentes consecutivas del vector y_s
$\mathbf{Z}$	Matriz de las variables predictoras
$\mathbf{z}_i^T$	Fila i de $\mathbf{Z}$
α	Factor de proporcionalidad de la función $p(c_{ij})$
α', β, d_i, d_j	Parámetros del problema de la distribución de tráfico
α^k	Parámetros del modelo logit
α_r	Coeficiente r de la función neuronal
$\boldsymbol{\beta}$	Vector de los coeficientes de regresión

$\hat{\boldsymbol{\beta}}$	Estimador de menor valor absoluto de $\boldsymbol{\beta}$
$\tilde{\boldsymbol{\beta}}$	Estimador Minimax de $\boldsymbol{\beta}$
β_a	Multiplicador de Lagrange asociado a a
	Restricción de máxima capacidad del arco
$\hat{\boldsymbol{\beta}}_{(i)}$	Estimador de mínimo valor absoluto de $\boldsymbol{\beta}$ cuando se omite la observación i-ésima
$\tilde{\boldsymbol{\beta}}_{(i)}$	Estimador Minimax de $\boldsymbol{\beta}$ cuando se omite la observación i-ésima
β_1	Parámetro del modelo logit
$\delta_{a,r}$	Dato binario que indica si el arco a pertenece a la ruta r
δ_s	Error asociado a la salida y_s
Δ	Salto
ϵ	Variable no negativa que representa el máximo error al aprender la opción 3
	Grado de incompatibilidad global de dos matrices de probabilidad condicional
$\phi_a(f_a)$	Función para modelar la capacidad del arco
$\phi_r(x)$	Funciones neuronales invertibles básicas
γ_{ij}	Importancia relativa del error en p_{ij}
$\boldsymbol{\lambda}$	Multiplicadores de Lagrange
$\boldsymbol{\lambda}$, $\boldsymbol{\psi}$	Multiplicadores de Lagrange
$\boldsymbol{\mu}$	Multiplicadores de Lagrange de las restricciones de reserva
Ω	Un par origen–destino
$\boldsymbol{\tau}$, $\boldsymbol{\eta}$	Vectores de las probabilidades marginales
τ_i, η_j	Componente i y j de los vectores $\boldsymbol{\tau}$ y $\boldsymbol{\eta}$, respectivamente
$\theta(.)$	Función dual
$\boldsymbol{\varepsilon}$	Vector de errores aleatorios

Capítulo 13

$\mathbf{A}_\pi$	Cono poliédrico convexo o cono generado por $\mathbf{A}$
$\mathbf{A}_\rho$	Espacio vectorial generado por $\mathbf{A}$
x^*	Variable no negativa
x^+,x^-	Partes positiva y negativa de la variable x, respectivamente
x_{n+1}	Variable artificial

Apéndice A

$\mathbf{A}_\pi$	Cono poliédrico convexo o cono generado por $\mathbf{A}$
$\mathbf{A}_\pi^*$	Dual no positivo del cono $\mathbf{A}_\pi$
$\mathbf{A}_\rho$	Espacio vectorial generado por $\mathbf{A}$
$\mathbf{A}_\rho^*$	Dual no positivo del espacio vectorial $\mathbf{A}_\rho$

$\mathbf{B}$	Generadores de la componente de espacio vectorial del cono $\mathbf{A}_\pi$
$\mathbf{C}$	Generadores de la componente propia de cono del cono $\mathbf{A}_\pi$
$\mathbf{C}_S$	Cono asociado al poliedro S
h	Contador de iteraciones del algoritmo Γ
$\mathbf{I}^-$	Índices asociados a los productos escalares no negativos
$\mathbf{I}^+$	Índices asociados a los productos escalares positivos
$\mathbf{I}^0$	Índices asociados a los productos escalares nulos
$\mathbf{I}_n$	Matriz identidad de dimensión n
K_{W^0}	Conjunto de todos los vectores de $\mathbf{W}^0$ cuya última componente es no nula
pivote	Índice de la columna pivote
$\mathbf{Q}_\lambda$	Politopo generado por los vectores columna de $\mathbf{Q}$
ρ	Número real no restringido
$\mathbf{V}^0_\rho$	Componente de espacio vectorial de $\mathbf{C}^{T^*}$
$\mathbf{V}$, $\mathbf{W}$	Matriz de los generadores del cono dual $\mathbf{A}^*_\pi$
$\mathbf{W}^0_\pi$	Componente propia de cono de $\mathbf{C}^{T^*}$
$\mathbf{W}^\pm$	Conjunto de vectores $\{\mathbf{w}\vert\mathbf{w} = t_j\mathbf{w}_i - t_i\mathbf{w}_j\}$
λ	Número real en el intervalo [0,1]
π	Número real no negativo

Bibliografía

[1] Arnold, B., Castillo, E., and Sarabia, J. M., *Conditional Specification of Statistical Models*, Springer-Verlag, New York, 1999.

[2] Arnold, B., Castillo, E., and Sarabia, J. M., *An Alternative Definition of Near Compatibility*, Technical Report 99-1, Univ. Cantabria, DMACC, 1999.

[3] Arnold, B., Castillo, E., and Sarabia, J. M., *Compatibility and Near Compatibility with Given Marginal and Conditional Information*, Technical Report 99-2, Univ. Cantabria, DMACC, 1999.

[4] Arnold, B., Castillo, E., and Sarabia, J. M., *Compatibility and Near Compatibility of Bayesian Networks*, Technical Report 99-3, Univ. Cantabria, DMACC, 1999.

[5] Arnold, B., Castillo, E., and Sarabia, J. M., *Exact and Near Compatibility of Discrete Conditional Distributions*, Technical Report, 00-1, Univ. of Cantabria, DMACC, 1999.

[6] Arthanari, T. S., and Dodge, Y., *Mathematical Programming in Statistics*, WIley, New York, 1993.

[7] Batut, J., Renaud, A., and Sandrin, P., "New Software for the Generation Rescheduling in the Future EDF National Control Center," *Proc. 10th Power Systems Computacional Conf.*, PSCC'90, Graz, Austria, 1990.

[8] Batut, J., and Renaud, A., "Daily Generation Scheduling Optimization with Transmission Constraints: a New Class of Algorithms," *IEEE Trans. Power Syst.* **7**(3), 982–989 (1992).

[9] Bazaraa M. S., Sherali, H. D., and Shetty C. M., *Nonlinear Programming, Theory and Algorithms*, 2nd ed., Wiley, New York, 1993.

[10] Beckman, M. J., and McGuire, C. B., *Studies in the Economics of Transportation*, Yale Univ. Press, New Haven, CT, 1956.

[11] Bertsekas, D. P., *Constrained Optimization and Lagrange Multiplier Methods*, Academic Press, New York, 1982.

[12] Bertsekas, D. P., Lauer, G. S., Sandell, N. R., and Posbergh, T. A., "Optimal Short-Term Scheduling of Large-Scale Power Systems," *IEEE Trans. Automatic Control*, **28**(1), 1–11 (1983).

[13] Bisschop, J., and Roelofs, M., *AIMMS, The User's Guide*, Paragon Decision Technology, 1999.

[14] Boyer, C. B., *A History of Mathematics*, Wiley, New York, 1968.

[15] Bradley, S. P., Hax, A. C., and Magnanti, T. L., *Applied Mathematical Programming*, Addison-Wesley, Reading, MA, 1977.

[16] Brannlund, H., Sjelvgren, D., and Bubenko, J., "Short Term Generation Scheduling with Security Constraints," *IEEE Trans. Power Syst.*, **1**(3), 310–316 (1988).

[17] Bronson, R., *Operations Research*, McGraw-Hill, New York, 1982.

[18] Canann, S. A., Muthukrishnan, S. N., and Phillips, R. K., "Topological Refinement Procedures for Triangular Finite Element Meshes," *Eng. Comput.*, **12**, 243–255 (1996).

[19] Canann, S. A., Tristano, J. R., and Staten, M. L., "An Approach to Combined Laplacian and Optimization-Based Smoothing for Triangular, Quadrilateral, and Quad-Dominant Meshes," *Proc. 7th International Meshing Roundtable*, 1998.

[20] Castillo, E., Fontenla, O., Guijarro, B., and Alonso, A., "A Global Optimum Approach for One-Layer Neural Networks," *Neural Computation*, (in press).

[21] Castillo, E., Cobo, A., Jubete, F., and Pruneda, E., *Orthogonal Sets and Polar Methods in Linear Algebra: Applications to Matrix Calculations, Systems of Equations and Inequalities, and Linear Programming*, Wiley, New York, 1999.

[22] Castillo, E., Hadi, A. S., and Lacruz, B., "Regression Diagnostics for the Least Absolute Value and the Minimax Methods," *Commun. Statist. Theor. Meth.* (in press).

[23] Castillo, E., and Mínguez, R., *An Optimization-Based Method for Mesh Generation and Smoothing*, Technical Report, 01-1, Univ. of Cantabria, DMACC, 2001.

[24] Chattopadhyay, D., "Application of General Algebraic Modeling System to Power System Optimization," *IEEE Trans. Power Systems* **14**(1), 15–22 (Feb. 1999).

[25] Charnes, A., and Cooper, W.,"Programming with Linear Fractional Programming," *Naval Research Logistics Quart.* **9**, 181–186 (1962).

[26] Clarke, F. H., *Optimization and Nonsmooth Analysis*, SIAM, Philadelphia, PA, 1990.

[27] Conn, A. R., Gould, N. I. M., and Toint, P. L., *LANCELOT: A FORTRAN Package for Large-Scale Nonlinear Optimization* (*Relase A*), No. 17 in Springer Series in Computational Mathematics, Springer-Verlag, New York, 1992.

[28] Cook, R. D., "Detection of Influential Observations in Linear Regression," *Technometrics* **19**, 15–18 (1977).

[29] Cuilliere, J. C., "An adaptive method for the Automatic Triangulation of 3D Parametric Surfaces," *Computer-Aided Design* **30**(2), 139–149 (1998).

[30] Dantzig, G. B., *Linear Programming and Extensions*, Princeton Univ. Press, Princeton, NJ, 1963.

[31] Davidon, W. C., *Variable Metric Method for Mininimization*, EC Research Development Report ANL-5990, Argonne National Laboratory, Chicago, IL, 1959.

[32] Dillon, T.S., Edwin, K.W., Kochs, H.D., and Tand, R.J., "Integer Programming Approach to the Problem of Optimal Unit Commitment with Probabilistic Reserve Determination," *IEEE Trans. Power Apparatus and Systems* **PAS-97**(6), 2154–2166 (1978).

[33] Du, D. Z., and Zhang, X. S., "Global convergence of Rosen's Gradient Projection Method," *Mathematical Programming* **44**, 357–366 (1989).

[34] Everett, H., "Generalized Lagrange Multiplier Method for Solving Problems of Optimum Allocation of Resources," *Operations Research* **11**, 399–417 (1963).

[35] Farkas, J., "Uber der einfachen Ungleichungen," *J. Reine und Angewandte Mathematik* **124**, 1–27 (1902).

[36] Farouki, R. T., "Optimal Paramaterizations," *Computer Aided Geometric Design*, **14**, 153-168 (1997).

[37] Ferreira, L. A. F. M., "On the Duality Gap for Thermal Unit Commitment Problems," *Proc ISCAS'93*, 2204–2207 (1993).

[38] Fiacco, A. V., and McCormick, G. P., *Nonlinear Programming: Sequential Unconstrained Minimization Techniques*, Wiley, New York, 1968.

[39] Fletcher, F., and Powell, "A Rapidly Convergent Descent Method for Minimization," *Comput. J.*, **6**, 163–168 (1963).

[40] Forrest, J. J. H., and Tomlin, J. A., "Implementing Interior Point Linear Programming Methods for the Optimization Subroutine Library," *IBM Syst. J.* **31**(1), 26–38 (1992).

[41] Fourer, R., Gay, D. M., and Kernighan, B. W., *A Modeling Language for Mathematical Programming*, The Scientific Press Series, 1993.

[42] Freitag, L. A., Jones, M., and Plassmann, P., "An Efficient Parallel Algorithm for Mesh Smoothing," *Proc. 4th Int. Meshing Roundtable*, 47–58 (1995).

[43] Freitag, L. A., "On Combining Laplacian and Optimization-Based Mesh Smoothing Techniques," **AMD-220**, Trends in Unstructured Mesh Generation, 37–43 (1997).

[44] Freitag, L. A., and Gooch, C. O., "Tetrahedral Mesh Improvement Using Swapping and Smoothing," *Int. J. Num. Meth. Eng.*, **40**, 3979–4002 (1997).

[45] Frisch, K. R., *The Logarithmic Potential Method for Convex Programming*, unpublished manuscript, Institute of Economics, Univ. Oslo, Norway, May 1955.

[46] García, R., López M. L., and Verastegui, D., "Extensions of Dinkelbach's Algorithm for Solving Nonlinear Fractional Programming Problems," *Top* **7**, 33–70 (1999).

[47] Garfinkel, R. S., and Nemhauser, G. L., *Integer Programming*, Wiley, New York, 1984.

[48] Gill, P. E., Murray, W., Saunders, M. A., Tomlin, J. A., and Wright, M. H., "On Projected Newton Barrier Methods for Linear Programming and an Equivalence to Karmarkar's Projective Method," *Math. Program.*, **36**, 183–209 (1986).

[49] Goldstein, A. A. "Convex programmingin Hilbert space," *Bull. Am. Math. Soc.* **70**, 709–710 (1964).

[50] Gomory, R. E., *An Algorithm for the Mixed Integer Problem*, Rand Report RM–25797., Rand, Santa Monica, CA, July 1960.

[51] Hiriart-Urruty, J. B., and Lemaréchal C., *Convex Analysis and Minimization Algorithms* Vols. I and II, Springer-Verlag, Berlin, 1996.

[52] Hobbs, W. J., Hermon, G., Warner, S., and Sheble, G. B., "An Enhanced Dynamic Programming Approach for Unit Commitment," *IEEE Trans. Power Systems* **3**(3), 1201–1205 (1988).

[53] *http://www.gams.com.*

[54] *http://www.ampl.com.*

[55] *http://www.aimms.com.*

[56] *IBM Optimization Subroutine Library (OSL)*, Release 2, Publication No. SC23-0519-03, 1992.

[57] Karmarkar, N. K., "A New Polynomial-Time Algorithm for Linear Programming," *Combinatorica*, **4**, 373–395 (1984).

[58] Kort, B. W., "Rate of Convergence of the Method of Multipliers with Inexact Minimization," in *Nonlinear Programming*, **2**, Academic Press, New York, 193–214, 1975.

[59] Jahn, J., *Introduction to the Theory of Nonlinear Optimization.* Springer-Verlag, Berlin, 1996.

[60] Jiménez Redondo, N., and Conejo, A. J., "Short-Term Hydro-Thermal Coordination by Lagrangian Relaxation: Solution of the Dual Problem," *IEEE Trans. Power Syst.*, **14**(1), 266–273 (1999).

[61] Jubete, F., *Programación no Lineal Convexa. El Método Polar*, CIS, Santander, Spain, 1987.

[62] Jubete, F., *El Cono Poliédrico Convexo. Su Incidencia en el Álgebra Lineal y la Programación no Lineal*, CIS, Santander, Spain, 1991.

[63] Khachiyan, L. G., "A Polynomial Algorithm in Linear Programming," *Doklady Akademii Nauk SSSR* No. 244, 1093–1096, [translated into English in Soviet Mathematics Doklady No. 20, 191–194 (1979)].

[64] Kuenzi, H. P., Tzchach, H. G., and Zehnder, C. A.,*Numerical Methods of Mathematical Optimization*, Academic Press, Orlando, FL, 1971.

[65] Lemaréchal, C., *Nondifferentiable Optimization*, in Optimization (G. L.Ñemhauser, A. H. G., Rinnooy and M. J., Tood, eds., Vol. 1 in *Handbooks in Operations Research and Management Science*, Elsevier, Amsterdam, 1989.

[66] Levitin, E. S., and Polyak, B. T., "Constrained Minimization Methods," *USSR Comput. Math. and Math. Phys.* **6**, 1–50 (1966).

[67] Li, T. S., McKeag, R. M., and Armstrong, C. G., "Hexahedral Meshing Using Midpoint Subdivision and Integer Programming," *Comput. Meth. Appl. Mech. Eng.*, **124**, 171–193 (1995).

[68] Luenberger, D. G., *Linear and Nonlinear Programming*, 2nd ed., Addison-Wesley, Reading, MA, 1989.

[69] Luh, P. B., Zhang, D., and Tomastik, R. N., "An Algorithm for Solving the Dual Problem of Hydrothermal Scheduling," *IEEE Trans. Power Syst.* **13**(2), 593–600 (1998).

[70] Medina, J., Quintana, V.H., Conejo, A.J., and Pérez Thoden, F., "A Comparison of Interior-Point Codes for Medium-Term Hydro-Thermal Coordination," *IEEE Trans. Power Syst.* **13**(3), 836–843 (1998).

[71] Medina, J., Quintana, V. H., and Conejo, A. J., "A Clipping-Off Interior-Point Technique for Medium-Term Hydro-Thermal Coordination," *IEEE Trans. Power Syst.* **14**(1), 266–273, (Feb. 1999).

[72] Meggido, N., "Pathway to the Optimal Set in Linear Programming," in *Progress in Mathematical Programming: Interior Point and Related Methods*, Springer Verlag, New York, 1989, pp. 131–158.

[73] Mendes, V. M., Ferreira, L. A. F. M., Roldao, P., and Pestana, R., "Optimal Short-Term Scheduling in Large Hydrothermal Power Systems," *Proc. 11th Power Systems Computation Conf. PSCC'93.* Vol. II, Avignon, France, 1993, pp. 1297–1303.

[74] Merlin, A., and Sandrin, P., "A New Method for Unit Commitment at Electricité de France," *IEEE Trans. Power Apparatus Syst.*, **PAS-102**(5), 1218–1225 (1983).

[75] Mehrotra, S., *On the Implementation of a (Primal Dual) Interior Point Method*, Technical Report 90–03, Dep. Industrial Engineering and Management Sciences, Northwestern Univ., Evanston, IL, 1990.

[76] Migdalas, A., "A Regularization of the Frank–Wolfe Method and Unification of Certain Nonlinear Programming Methods," *Math. Program.* **65**, 331–345 (1994).

[77] Muckstadt, J. A., and Koening, S. A., "An Application of Lagrangian Relaxation to Scheduling in Power Generation Systems," *Operation Research* **25**(3), 387–403 (1977).

[78] Murtagh, B. A., and Saunders, M. A., *MINOS 5.0 Users's Guide*, Report SOL 83–20, Dep. Operations Research, Stanford Univ.

[79] Nemhauser, G. L., and Wolsey, L. A., *Integer and Combinatorial Optimization.* Wiley, New York, 1988.

[80] Nguyen S., and Dupis, C., "An Efficient Method for Computing Traffic Equilibria in Networks with Asymmetric Transportation Costs," *Transportation Science*, **18**, 185–202 (1984).

[81] Nocedal, A. J., and Wright, J., *Numerical Optimization*, Springer, New York, 1999.

[82] Ortúzar, J. de D., and Willumsen, L. G., *Modelling Transport*, Wiley, New York, 1994.

[83] Patrikson, M., "Partial linearization methods in nonlinear programming," *J. Optim. Theor. and Appl.* **78**, 227–246 (1993).

[84] Patriksson, M., *The Traffic Assignment Problem. Models and Methods*, VSP, Utrecht, The Netherlands, 1994.

[85] Pellegrino, F., Renaud, A., and Socroun, T., "Bundle. Augmented Lagrangian Methods for Short-Term Unit Commitment," *Proc. 12th Power Systems Computation Conf. PSCC'96*, Vol. II, 730–739. Dresden, Germany, 1996.

[86] Polak, E., *Optimization: Algorithms and Consistent Approximations*, Springer, New York, 1997.

[87] Polyak B. T., *Introduction to Optimization*, Optimization Software, New York, 1987.

[88] Potts, R., and Oliver, R., *Flows in Transportation Networks*, Academic Press, New York, 1972.

[89] Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P., *Numerical Recipes in C. The Art of Scientific Computing*, Cambridge Univ. Press, New York, 1985.

[90] Torres, G. L., and Quintana, V. H., *Power System Optimization via Successive Linear Programming with Interior Point Methods*, Univ. of Waterloo. Dep. Electrical and Computer Engineering, Technical Report UW E&CE 96–02, Jan. 1996.

[91] Rakic, M. V., and Marcovic, Z. M., "Short Term Operation and Power Exchange Planning of Hydro-Thermal Power Systems," *IEEE Trans. Power Syst.*, **9**(1), 359–365 (1994).

[92] Rao, S. S., *Optimization Theory and Applications.* 2nd ed., Wiley Eastern Limited, New Delhi, 1984.

[93] Rao, S. S., *Engineering Optimization. Theory and Practice*, 3rd ed., Wiley, New York, 1996.

[94] Renaud, A., "Daily Generation Management at Electricité de France: from Planning towards Real Time," *IEEE Trans. Automatic Control* **38**(7), 1080–1093 (1993).

[95] Rockafellar, R. T., *Convex Analysis*, Princeton Univ. Press, Princeton. NJ, 1970.

[96] Rosen, J. B., "The gradient projection method for nonlinear programming," Parts I and II, *J. Soc. Ind. Appl. Math.*, **8**, 181–217 (1960); **9**, 514–532 (1961).

[97] Shimizu, K., Ishizuka, Y., and Bard, J. F., *Nondifferentiable and Two-Level Mathematical Programming*, Kluwer Academic Publishers, Boston MA, 1997.

[98] Sheffi, Y., *Urban Transportation Networks: Equilibrium Analysis with Mathematical Programming Methods*, Prentice-Hall, Englewood Cliffs, NJ, 1984.

[99] Troutman, J. L., *Variational Calculus and Optimal Control: Optimization with Elementary Convexity*, 2nd ed., Springer, New York, 1996.

[100] Vanderbei, R. J., "Interior Point Methods: Algorithms and Formulations," *ORSA J. Comput.*, **6**(1), 32–34 (1994).

[101] Wang, S. J., Shahidehpour, S. M., Kirschen, D. S., Mokhtari, S., and Irisarri, G. S., "Short-Term Generation Scheduling with Transmission and Environmental Constraints using an Augmented Lagrangian Relaxation," *IEEE Trans. Power Syst.*, **10**(3), 1294–1301, 1995.

[102] Wardrop, J. G., "Some Theoretical Aspects of Road Traffic Research," *Proc. Institute of Civil Engineers Part II*, 325–378 (1952).

[103] Wolfram, S., *The Mathematica Book*, Wolfram Media and Cambridge Univ. Press, Cambridge, UK, 1996.

[104] Wood, A. J., and Wollenberg, B. F., *Power Generation Operation and Control*, 2nd ed., Wiley, New York, 1996.

[105] Wright, S. J., *Primal-Dual Interior-Point Methods*, SIAM, Philadelphia, 1997.

[106] Yan, H., Luh, P. B., Guan, X., and Rogan, P. M., "Scheduling of Hydrothermal Power Systems," *IEEE Trans. Power Syst.* **8**(3), 1135–1365 (1993).

[107] Zhuang, F., and Galiana, F. D., "Toward a More Rigorous and Practical Unit Commitment by Lagrangian Relaxation," *IEEE Trans. Power Syst.* **3**(2), 763–773 (1988).

Índice de Materias

Óptimo
global ejemplo, 208
global y local, 212

Academia de Ingeniería, 29, 31
Algoritmo
de optimización sin restricciones, 240
gamma, 495
modificado del ascenso más rápido, 261
MPE, 149
Aplicaciones, 379
a inteligencia artificial, 379
a la probabilidad, 395
al CAD, 388
Armador, problema del, 26

Búsquedas lineales
con derivadas, 240, 241
sin derivadas, 242
Braquistocrona, 413

caracterización de puntos extremos
teorema, 84
Caso
compatible ejemplo, 398
con demanda elástica, 443
incompatible ejemplo (con ceros), 400
Casos especiales, 198
ejemplos, 218
Casos particulares, 232
Centrales térmicas, programación horaria de, 39, 43
Combinación lineal
convexa definición, 112
definición, 108
no negativa, 109
Compatibilidad, 489
ejemplo, 502
matrices de probabilidad condicional, 396
sistemas lineales, 501
Condiciones
de Karush–Kuhn–Tucker, 195
optimalidad, 194
suficiencia, 216, 217
de optimalidad
suficiencia y convexidad, 211
de optimalidad en programación no lineal, 187
de regularidad, 232
necesarias de optimalidad, 193
Conjunto
convexo, 101
convexo definición de, 103
de soluciones factibles, 99
de todas las soluciones, 489
Conjuntos poliédricos
representación, 114
Cono
asociado a un poliedro, 492
dual, 491
dual ejemplo, 496
forma general, 110
generadores, 110
polar, 490
poliédrico convexo, 109
poliédrico convexo definición, 110
representación, 111
vectores en un, 491
Cono dual, 490

Cono polar definición, 490
Control óptimo
  de un oscilador armónico, 423
  para el alcance de un blanco, 420
Convexa
  combinación lineal, 112
  definición, 212
Convexidad, 211
  caracterización de óptimo global, 216
  caracterización de funciones, 215
Convexidad y diferenciabilidad
  teorema, 214
Convexo
  conjunto, 101
  cono, 104
Coordinación hidrotérmica a corto plazo, 453
Cuasi-compatibilidad, 400
  ejemplo, 401, 403
Cubo, ejemplo del, 101
Cuerda colgante, 415

Definición
  óptimo global y local, 212
  combinación lineal, 108
  combinación lineal convexa, 112
  conjunto convexo, 103
  cono polar, 490
  cono poliédrico convexo, 110
  de restricción activa, 206
  dirección, 107
  dirección extrema, 107
  dual no positivo, 490
  espacio vectorial, 108
  forma general de un cono, 110
  función convexa, 212
  matriz básica, 83
  poliedro, 113
  politopo, 112
  punto extremo, 105
  restricción activa, 206
  solución factible, 76
Despacho económico, 336
Dieta
  problema de la, 520
Diferenciabilidad, 193
Dirección
  definición, 107
  extrema definición, 107
Direcciones de descenso
  algoritmo, 247
Discretización de problemas, 410
Distribución
  de viajes, 64
Dual no positivo
  definición, 490
Dualidad, 86

Ejemplo
  caso compatible, 398
  caso incompatible (con ceros), 400
  cuasi-compatibilidad, 401, 403
  de búsquedas lineales con interpolación cuadrática, 244
  de condiciones de KKT, 203
  de la bombilla, 48
  de la columna, 53, 358
  de la estructura con dos barras, 357
  de la lámpara, 355
  de la mochila, 339
  de la superficie, 50, 355
  de la tienda, 48, 354
  de sistema de inecuaciones, 505
  de vigas y cuerdas, 334
  del paquete postal, 47, 353
  del transporte de arena, 50, 356
  del voladizo, 357
  método de ramificación y acotación, 167, 171
  optimización sin restricciones, 252
  restricciones alternativas, 471
Ejemplos
  de programación lineal, 319
  de programación no lineal, 353
  en GAMS, 319
  en ingeniería eléctrica, 56
  geométricos, 47
  mecánicos, 51

Elementos de un PPL, 3
Envolvente convexa, 112
Espacio vectorial, 108
    como cono, 111, 468, 469
    definición, 108
    representación, 109
Estimación
    del estado de un sistema eléctrico, 56, 361
    del estado de una red, 58
Estructura con dos barras, 52
Existencia
    de soluciones óptimas, 192

Factibilidad
    región de, 76
Factible
    región, 143
    solución, 76
Farkas
    lema, 491
Flujo
    de potencia óptimo, 58, 61
Forma estándar, 82
Forma general de un cono
    definición, 110
Función
    dos veces diferenciable, 214

GAMS
    asteriscos, 310
    bucles, 313
    características del lenguaje, 297
    comando display, 311
    comando solve, 307
    comandos condicionales, 311
    conjuntos dinámicos, 313
    ecuaciones, 306
    ejemplo ilustrativo, 292
    escalares, 300
    expresiones matemáticas, 302
    ficheros de salida, 316, 317
    introducción, 291
    modelos, 306
    parámetros, 300
    software, 291
    tablas, 300
    trucos, 476
    variables, 303

Hiperplanos, 104
Holgura dual, 279
Horario
    problema del, 34
Horario problema del, 32

Identificación
    de síntomas relevantes, 27, 28, 341
    de subespacios, 108
Infactible
    región, 143

Lema
    de dualidad débil, 89
    Farkas, 491
Localización de plantas industriales, 35, 37

Método
    barrera algoritmo, 271
    barrera ejemplo, 271, 272
    de direcciones conjugadas, 257
    de los cortes de Gomory
        algoritmo, 177, 179
        generación de cortes, 175
        método, 175
    de Newton, 254
    de penalización exterior
        algoritmo, 268
        convergencia, 267
        ejemplo, 268
    de punto exterior
        selección de la variable entrante $\mathbf{x}_N$, 148
    de ramificación y acotación, 167, 171
    del gradiente, 252
    del mínimo valor absoluto, 405
    del punto exterior, 143
        detección de infactibilidad y no acotación, 146

ejemplo no acotado, 156
ejemplo no factible, 157
ejemplos, 150
fase de iteraciones estándar, 147
fase inicial, 145
fase reguladora, 146
primer ejemplo, 151
segundo ejemplo, 153
selección de la variable saliente $\mathbf{x}_N$, 148
del punto interior
actualización del parámetro de penalización, 280
cotas, 281
criterio de parada, 280
ejemplo, 276
fundamentación, 275
infactibilidad inicial, 279
selección del punto inicial, 282
del punto interior en programación lineal, 274
minimax, 405
quasi-Newton, 241, 255
simplex, 120, 143
detección de la no acotación, 128
detección de la no factibilidad, 129
ejemplo no acotado, 140
ejemplo no factible, 141
ejemplos, 133
etapa de iteraciones estándar, 129
identificación de una solución óptima, 126
iteración reguladora, 128
operación elemental de pivote, 125
selección de la variable entrante en $\mathbf{x}_B$, 130
selección de la variable saliente $\mathbf{x}_B$, 131
simplex revisado, 131
Métodos
barrera, 269
barrera convergencia, 271
de búsqueda lineal, 240
de penalizaciones, 266
de penalizaciones exteriores, 266
duales, 259
Mínimo
global, 193
local, 193
Mínimos cuadrados, 405
ejemplo de ajuste, 219
Matrices
de probabilidad general compatibilidad, 396
Matriz
básica definición, 83
semidefinida positiva, 215
Mochila
problema, 25
Modelo
de localización de plantas industriales, 35
de variables binarias, 164
Modelos
de asignación con restricciones adicionales, 440
de localización de plantas productivas, 347
de regresión, 404
Multiplicadores
significado económico, 459
técnicas de actualización, 458

No acotados
problemas, 115

Obtención del dual a partir del primal en forma estándar, 87
Obtención del problema dual, 88
Operación de pivotación, 125
Optimización
con restricciones algoritmos, 259
sin restricciones, 246

Planificación
de la producción, 512, 513
Poliedro

definición, 113
Politopo
definición, 112
representación, 112
Potencia eléctrica, 18
PPL en forma estándar, 80
PPLEM, 163, 339
otros problemas, 174
Problema
de asignación de tráfico, 432
de Cobb–Douglas, 220
de flujo en redes, 11, 12, 327
de la academia de ingeniería, 342
de la asignación de tráfico, 66, 68
de la cartera de valores, 13, 15, 332
de la dieta, 9, 10, 325
de la matriz equilibrada, 62, 369
de la planificación de la producción, 6, 8, 323
de la red de abastecimiento de agua, 329, 367
de las redes, 521
de matrices equilibradas, 263
de programación lineal, 76
del armador, 26
del despacho económico, 18, 21
del horario, 345
del transporte, 4, 5, 319
ejemplo, 508
dual, 86, 88
dual del carpintero, 90
Problema dual
ejemplo, 226
Problemas acotados, 115
Procedimiento gamma, 495
Programación
fraccional, 467
horaria de centrales térmicas, 349, 453
lineal definición, 76
lineal entera mixta, 25
lineal, introducción, 75
no lineal, 47
Programación lineal, 3
teorema fundamental, 84
Punto extremo
definición, 105

Ramificación y acotación
algoritmo, 165
estrategias, 166
método, 165
Red de comunicaciones, 93
Red de tráfico, 428
Redes neuronales y funcionales, 379
Regla
de Armijo, 248
Relajación lagrangiana, 454
Reparto
óptimo de cargas, 364
Representación
de poliedros, 114
de poliedros teorema, 115
de un cono, 111
de un politopo, 112
Resolución
del problema de programación lineal, 119
sistemas lineales, 502
Restricciones alternativas
ejemplo, 471

Semiespacios, 104
Sensibilidades, 85
Sistema
de inecuaciones, 504
de vigas y cuerdas, 15, 17, 359
homogéneo de ecuaciones
solución, 109
lineal homogéneo de desigualdades, 111
Sistemas
de transporte, 426
Solución
óptima, 76
única, 77
infactible, 78
no acotada, 78
Soluciones

básicas, 83
múltiples, 77
Suficiencia
de las condiciones de Karush–Kuhn–Tucker (KKT), 217

Tablas input–output, 518
Teoría de dualidad, 222
Teorema
caracterización de puntos extremos, 84
convergencia, 275
convexidad y diferenciabilidad, 214
de dualidad, 90
para problemas convexos, 225
de dualidad débil, 224
diferenciabilidad de la función dual, 261
discontinuidad, 473
dualidad débil, 224
espacio vectorial como cono, 469
función no convexa a trozos, 474
funciones convexas, 215
funciones no convexas a trozos, 474
fundamental de la programación lineal, 84
identificación de subespacios, 108
representación de conjuntos convexos finitos, 107
representación de poliedros, 115
representación de un espacio vectorial, 109
representación de un politopo, 112
restricciones alternativas, 470
Transformación de pivotación, 125
Teoremas
de dualidad, 89
Transformación a la forma estándar, 81
Transporte
problema del, 507
Trucos, 461, 476
estados inicial y final, 481
adición iterativa de restricciones a un problema, 480
análisis de sensibilidad, 482
de modelado, 461
definición de una matriz dispersa, 478
dependencia del flujo del programa, 482
descomposición de un problema separable, 479
desigualdades como igualdades, 463, 464
espacio vectorial como un cono, 468
función máximo, 466
funciones no continuas, 472
funciones no convexas a trozos, 473
funciones no lineales como lineales, 465
genéricos, 461
igualdades como desigualdades, 464
máximos como mínimos, 465
matriz simétrica, 477
problema de programación lineal fraccional, 467
restricciones alternativas, 470
restricciones condicionales, 472
valores absolutos, 466
variables no acotadas, 462
variables no negativas, 462, 463
variables no restringidas como no negativas, 462

Variables no negativas
forma estándar, 82
Variables no restringidas en signo mediante variables no negativas, 81
Vector en un cono, 491
Vigas y cuerdas, 54
Voladizo, 51